

# HIV-1 Reverse Transcriptase Inhibitor Design Using Artificial Neural Networks

Igor V. Tetko,<sup>†</sup> Vsevolod Yu. Tanchuk,<sup>†</sup> Neliya P. Chentsova,<sup>‡</sup> Svetlana V. Antonenko,<sup>‡</sup> Gennady I. Poda,<sup>†</sup> Valery P. Kukhar,<sup>†</sup> and Alexander I. Luik<sup>\*†</sup>

Biomedical Department, Institute of Bioorganic and Petroleum Chemistry, Murmanskaya, 1, Kiev-94, 253660, Ukraine, and Institute of Epidemic and Infectious Diseases, Spusk Stepana Razina 4, Kiev-38, 252038, Ukraine

Received November 30, 1993<sup>⊙</sup>

Artificial neural networks were used to analyze and predict the human immunodeficiency virus type 1 reverse transcriptase inhibitors. The training and control sets included 44 molecules (most of them are well-known substances such as AZT, dde, etc.). The activities of the molecules were taken from literature. Topological indices were calculated and used as molecular parameters. The four most informative parameters were chosen and applied to predict activities of both new and control molecules. We used a network pruning algorithm and network ensembles to obtain the final classifier. Increasing of neural network generalization of the new data was observed, when using the aforementioned methods. The prognosis of new molecules revealed one molecule as possibly very active. It was confirmed by further biological tests.

## Standard Neural Network with Back-Propagation Algorithm

We used back-propagation neural networks (BPNN) trained by  $\delta$ -rule as the pattern recognition method.<sup>1</sup> Shown in Figure 1 is a typical neural network. The neurons are designated as circles. The number of layers  $n$  is arbitrary (usually  $n = 3$ ). The data are input to A, transformed on hidden layers, and output to B. Each input layer node corresponds to a single independent variable. Similarly, each output layer node corresponds to a different dependent variable. Each neuron value  $O_j$  ranging from 0 to 1 is calculated by eq 1,

$$O_j = 1/(1 + e^{-\lambda y_j}) \equiv f(y_j), y_j = \sum w_{ij}^s O_j' - \theta_j \quad (1)$$

where  $O_j'$  are neuron values at the  $n - 1$  layer,  $w_{ij}^s$  is the weight of the bond connecting the  $i$ th neuron in layer  $s$  and the  $j$  neuron in the next layer,  $\theta_j$  is a threshold value for neuron  $j$ , and  $\lambda$  is a parameter that expresses the nonlinearity of the neuron's operation. Usually  $\lambda$  and  $\theta_j$  are the same for all neurons in a layer. Neural network training is achieved by minimizing an error function,  $E_{gl}$ , with respect to the bond weights  $w_{ij}^s$  until its value becomes small enough (usually 0.01–0.1):

$$E_{gl} = E_{gl}(w_{ij}) = \sum_p \sum_k (O_k - t_k)^2 \quad (2)$$

where the inner summation is over all neurons that are considered as output units of the net,  $t_k$  is the desired output upon presentation of pattern  $p$ , and the outer sum is over patterns of the training set. A generalized  $\delta$ -rule has been used. In this algorithm, bond weights  $w_{ij}^s$  starting from small random values are changed by a gradient descent method during the training process. In these equations,  $\epsilon$  is a constant called the learning rate and  $\eta$  is a momentum rate. The last constant is used to avoid biases in a network during learning. Once the training is completed, weights are then held fixed

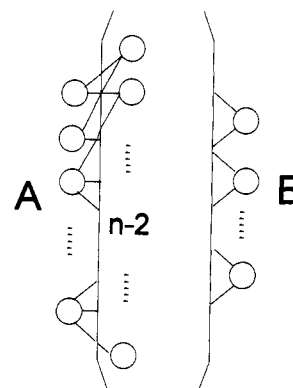


Figure 1. Typical neural network.

$$w_{ij}^s(t) = w_{ij}^s(t-1) + \Delta w_{ij}^s(t) \quad (3)$$

$$\Delta w_{ij}^s(t) = \epsilon \frac{\partial E_{gl}}{\partial w_{ij}^s(t-1)} + \eta \Delta w_{ij}^s(t) \quad (4)$$

for the testing mode of network operation. We used batch-training, i.e., weight updating was after presentation of all training patterns.

One of the main drawbacks of BPNN is an overfitting problem.<sup>2–4</sup> There is empirical evidence that generalization to novel input patterns is improved by using hidden layers with a small number of nodes. In these cases, generalization from the training set to novel inputs was better when the number of hidden nodes was relatively small. A small hidden layer forces the input patterns to be mapped through a low-dimensional space, enforcing proximities between hidden layer representations that were not necessarily present in the input pattern representations. Only the differences between patterns that are most important for decreasing error will be preserved as large distances between hidden layer patterns. Differences between input patterns that are not preserved in the hidden layer representation are thereby generalized over completely. Theoretical and empirical results regarding learnability, generalization, and network size can be found in refs 5 and 6.

The theoretical architecture for practical tasks of networks is generally unknown. One would rather

\* Author to whom all correspondence should be directed. e-mail: ibpc@bioorganic.kiev.ua.

<sup>†</sup> Institute of Bioorganic and Petroleum Chemistry.

<sup>‡</sup> Institute of Epidemic and Infectious Diseases.

<sup>⊙</sup> Abstract published in *Advance ACS Abstracts*, June 1, 1994.

overestimate the network size than underestimate it. Algorithms that adapt the network architecture during training and pruning redundant nodes (i.e., net pruning algorithm) have been developed.<sup>7-9</sup> These algorithms eliminate the need to guess an appropriate, initial network architecture prior to training. We have recently proposed a new simple algorithm.<sup>10</sup> A simulation on three different tasks shows its efficacy for determining the theoretically minimal architecture of networks during learning.<sup>10</sup> One of the main advantages of the algorithm is that it can evaluate the performance of input parameters during training and choose the most important ones. The last property is very important in QSAR and SAR studies where *a priori* is hard to evaluate the performance of molecular features. Here is a brief description of the algorithm.

### Pruning Algorithm

We estimated the importance of a neuron (in hidden or input layers) according to its sensitivity:

$$S_i = \sum_j \frac{(w_{ij}^s)^2}{(W_j^s)^2} = \sum_j \frac{(w_{ij}^s)^2}{(\max_k |w_{kj}^s|)^2}$$

The neuron having the greatest value  $S_i$  exerts the most significant influence on all other neurons in the next layer and vice versa. So, the importance of a node is measured by how much the node is relied upon by higher layer nodes. The importance of this definition of the neuron sensitivity is that it can be applied to evaluate not only neurons in hidden layers but also input parameters. It can be used after completing the network training to prune redundant neurons. To force a network to select the most important nodes, we add the cost of all neurons into a global error function:

$$E = E_{gl} + E_{cost}$$

$$E_{cost} = \frac{\alpha}{2} (\sum_i S_i - N)$$

where  $\alpha$  is the normalization coefficient and  $N$  is the number of all neurons except the input layer neurons. Extracting  $N$  helps us analogously normalize  $E_{cost}$  for networks with different numbers of neurons and visualize the learning process. Adding of  $E_{cost}$  results in adding the next terms into the learning rule for weights (if the  $\delta$ -rule is used):

$$\Delta w_{ji}^s = (\Delta w_{ji}^s)_{old} - \begin{cases} \alpha \frac{w_{ji}^s}{(W_i^s)^2}, & \text{if } w_{ji}^s \neq W_i^s \\ -\alpha \sum_{k \neq i} \frac{(w_{ki}^s)^2}{(W_i^s)^3}, & \text{if } w_{ji}^s = W_i^s \end{cases}$$

We chose  $\alpha$  so that  $E_{cost}$  was less in order than the desired error of the network  $\Delta E = 0.1$ . So, when training is in progress and  $E_{gl}$  is large, the cost of nodes is relatively small and becomes important only at the end of training.

When the training is near completion ( $E \approx 2\Delta E$ ), we inspect all neurons and delete a neuron having the least sensitivity. If the neuron is redundant, the error  $E$  first increases but then, within 10–300 epochs, the network retraining itself. Retraining is fast due to the correct structure of the network, which has been formed by the previous learning. After the retraining of the network, we repeat our pruning. Conversely, if pruning is unsuccessful, we restore the former network, continue

training for 200–1000 cycles, and repeat the network pruning. If even in this case pruning is impossible, we treat our network as final.

However, even the use of networks with the smallest architecture may result in an ambiguous generalization for new input patterns. The method of neural ensembles is a standard algorithm that removes such drawbacks.<sup>11</sup> We determined the level  $p$  of significance of the new molecule classification for given classes as described earlier.<sup>4</sup>

**Learning and Control Sets of Molecules.** Forty-four inhibitors of human immunodeficiency virus type 1 reverse transcriptase (HIV-1 RT) were taken from literature<sup>12-19</sup> as learning (30 compounds) and control (consists of 14 compounds randomly chosen, 10, 19, 26, 30, 35, 38, and 42, and taken from other sources, 12–16,<sup>19</sup> 43, and 44<sup>17</sup>) sets. The activity of compounds was rated for two classes: active and inactive compounds, according to their activity. Compounds with a ratio between their  $ED_{50}$  and the  $ED_{50}$  of AZT more than  $10^3$  were considered inactive. Twenty new molecules, synthesized and courteously given to us by Visnevskii et al.,<sup>20</sup> were evaluated as HIV-1 RT inhibitors.

**Parameters Used To Represent Molecules.** A set of about 50 topological indexes served as the input set. It was impossible to use all those parameters. Had we used all of them, the number of parameters would have been greater than the number of input patterns. In such cases, a network is trained very quickly but data generalization is rather poor because of overfitting. On the other hand, it was impossible to use the proposed here pruning algorithm because it would have taken a lot of time to prune such a large network. That is why we used another method for preliminary evaluation of input parameters. All parameters were scaled to unit variance and subjected to hierarchical cluster analysis. We used distances measured according to the median distance method in the space with Euclidean metrics.

This analysis divided all input parameters into six clusters that did not overlap with each other. From each cluster, the parameter with maximum correlation with the vector of molecular activities from the learning set was taken to be used in BPNN's training. Three of six used parameters are our own modifications of the Kier's index of molecular paths  $h\chi$  form:<sup>22</sup>

$$h\chi(G) = \sum (v_1 v_2 \dots v_{h+1})^{-1/2}$$

Here, summation must be done for all paths of length  $h > 1$ ;  $v_1 \dots v_{h+1}$  are the degrees of vertexes along a given path.

(1) This index is calculated by the same equation, but only the shortest paths with maximum products of vertex degrees are considered. This means that only the shortest paths between vertexes are considered and if there are several equally short paths between them the one that gives maximum product of vertex degrees must be chosen.

(2) The same with respect to minimum products minus first index.

Three other indexes are based on the connectivity matrix introduced by Barrish et al.<sup>23</sup> This method of calculating the connectivity matrix takes into account atom types. Elements of such a matrix are derived from the elements of the ordinary connectivity matrix by:

$$A'(i,j) = \frac{1}{n} \frac{36}{Z_i Z_j} \text{ if } A(i,j) = 1, \text{ else } A'(i,j) = 0$$

when  $n$  represents the bond order and  $Z_i$  and  $Z_j$  are the numbers of electrons of the  $i$ th and  $j$ th atoms.

A distance matrix based on this principle has also been proposed

$$D'(i,j) = \sum A'(k,l)$$

where  $k$  and  $l$  represent a pair of adjacent atoms lying on the shortest path from the  $i$ th vertex to  $j$ th. All such pairs are taken into account for a path. If there are several equally short paths, the path with minimal sum will be considered.

Table 1. Structure, Descriptors, and Observed Activities of Molecules

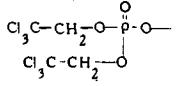
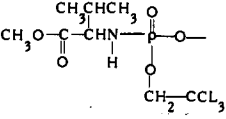
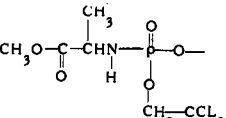
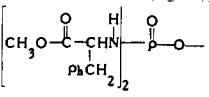

















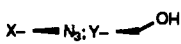


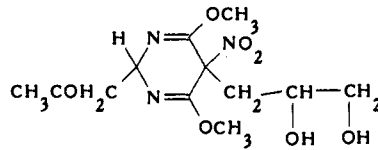
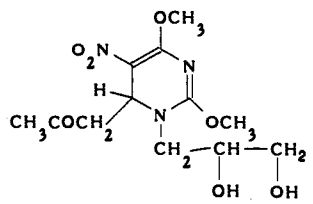
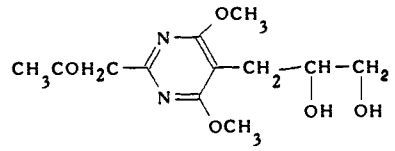
no.	substituent	used parameters						exp <sup>a</sup>	ref
		1	2	3	4	5	6		
Structure I									
1	X,Y-H; R-OH	0.01952	0.941	0.0654	0.01292	14.201	7.117	+	16
2	X,Y-H; R-NH <sub>2</sub>	0.0195	0.940	0.0642	0.01288	14.234	7.117	+	16
3	X-F; Y-H; R-NH <sub>2</sub>	0.0189	0.990	0.0631	0.01182	14.241	6.777	+	16
4	X-F; Y-H; R-OH	0.0189	0.990	0.0642	0.01186	14.209	6.777	+	16
5	X-H; Y-F; R-NH <sub>2</sub>	0.0189	0.990	0.0631	0.01182	14.241	6.777	+	16
Structure II									
6	X,Y-H; R-N <sub>3</sub> ; Z-CH <sub>3</sub>	0.000	1.075	0.0208	0.01301	19.773	6.473	+	12
7	X,Y,R-H; Z-CH <sub>3</sub>	0.000	0.993	0.0208	0.01773	16.720	5.500	+	12
8	X-F; Y,R,Z-H	0.0156	0.989	0.0548	0.01778	17.079	5.500	-	13, 14
9	X,R-F; Y,Z-H	0.0152	1.036	0.0544	0.01618	17.528	5.823	-	13
10 <sup>b</sup>	X-F; Y,R-H; Z-CH <sub>3</sub>	0.000	1.044	0.0074	0.01623	16.716	5.823	-	13
11	X-F; Y-H; R-N <sub>3</sub> ; Z-CH <sub>3</sub>	0.000	1.116	0.0070	0.01229	19.351	6.800	-	14
12 <sup>b</sup>	X,Y-H; R-F; Z-CH <sub>3</sub>	0.000	1.026	0.0214	0.01826	15.786	5.625	+	19
Structure III									
13 <sup>b</sup>		0.000	1.23927	0.0191	0.00661	27.095	8.666	+	19
14 <sup>b</sup>		0.000	1.267	0.0167	0.00557	28.496	9.969	-	19
15 <sup>b</sup>		0.000	1.240	0.0176	0.00615	27.975	9.258	-	19
16 <sup>b</sup>		0.0042	1.280	0.0217	0.00206	42.410	15.727	-	19
Structure IV									
17	X,Z-F; R,Y-H	0.000	1.044	0.0141	0.01648	16.515	5.823	-	13
18	X,R-F; Y,Z-H	0.000	1.039	0.0370	0.01395	44.197	6.444	+	13
19 <sup>b</sup>	X-F; R,Y,Z-H	0.0156	0.989	0.0293	0.01786	16.990	5.500	+	13
20	X,R,Z,Y-H	0.0156	0.989	0.0293	0.01786	16.990	5.500	+	13
Structure V									
21	Y-H	0.0158	0.934	0.0302	0.01993	16.955	5.533	+	14
22	Y-F	0.0156	0.989	0.0301	0.01831	16.787	5.500	+	13
Structure VI									
23	Y-H; Z-CH <sub>3</sub>	0.000	0.993	0.0076	0.01799	16.648	5.500	+	12
24	Y-F; Z-CH <sub>3</sub>	0.000	1.044	0.0075	0.01660	16.530	5.823	-	13
25	Z-H; Y-F	0.0156	0.989	0.0568	0.01823	16.876	5.500	-	13
Structure VII									
26 <sup>b</sup>	X=O; Y- 	0.01442	1.066	0.0634	0.01462	26.053	6.111	+	15
27	X-  ; Y- 	0.0172	1.113	0.0702	0.01063	45.111	7.095	-	15
28	X-  ; Y- 	0.0144	1.066	0.0515	0.01448	26.365	6.111	-	15
29	X-  ; Y- 	0.0169	1.092	0.0976	0.01189	39.193	6.900	-	15
30 <sup>b</sup>	X-  ; Y- 	0.0172	1.120	0.0637	0.00923	55.079	7.454	-	15
Structure VIII									
31	X=O; Y- 	0.01442	1.066	0.0634	0.01462	26.053	6.111	+	15
32	X-  ; Y-H	0.0152	1.036	0.0287	0.01624	17.443	5.823	-	15
33	X=O; Y-H	0.000	1.013	0.0255	0.01812	27.259	5.500	+	15
34	X-  ; Y-H	0.000	1.061	0.0273	0.01220	51.220	6.473	-	15
35 <sup>b</sup>	X-  ; Y- 	0.0172	1.113	0.0702	0.01063	45.111	7.095	-	15
36	X-  ; Y-H	0.000	1.013	0.0225	0.01791	27.674	5.500	-	15
37	X-  ; Y- 	0.0144	1.066	0.0515	0.01448	26.365	6.111	-	15

Table 1. (Continued)

no.	substituent	used parameters						exp <sup>a</sup>	ref
		1	2	3	4	5	6		
Structure VIII (Continued)									
38 <sup>b</sup>	X-  ; Y- 	0.0169	1.092	0.0976	0.01189	39.193	6.900	-	15
39	X- NHAc; Y-H	0.000	1.061	0.0265	0.01200	52.219	6.473	-	15
40	X- NHAc; Y- 	0.0172	1.120	0.0637	0.00923	55.079	7.454	-	15
Structure IX									
41	X-O	0.0068	1.170	0.0349	0.00899	17.111	7.380	+	18
42 <sup>b</sup>	X-S	0.0068	1.170	0.0342	0.00903	17.069	7.380	+	18
Structure X									
43 <sup>b</sup>	X-NH	0.0106	1.124	0.0410	0.00671	25.716	8.130	+	17
44 <sup>b</sup>	X-CH <sub>2</sub>	0.0106	1.124	0.0388	0.00645	26.961	8.130	-	17
New Compounds									
B1		0.0107	1.204	0.0285	0.0154	24.500	6.812		20
B2		0.0160	1.191	0.108	0.0154	17.832	7.364		20
B3		0.0077	1.110	0.0337	0.0191	23.115	6.368		20

<sup>a</sup> The rated activity of compounds: +, active compounds; -, inactive compounds. <sup>b</sup> This compound was used in the control set.

(3) The third index is the analog of index 2, but the degrees of vertexes are calculated using the described above connectivity matrix  $A'$ .

The described above indexes in our opinion somehow reflect the symmetry of molecules, although it is not known exactly how. Besides the analogs of  $h\chi$ , the optimized parameter set included three other indexes.

(4) The analog of Balabans<sup>24</sup> is as follows:

$$J(G) = \frac{2}{\mu + 2} \sum_{\text{adj}} (V_{D,i} V_{D,j})^{-1/2}$$

where

$$V_{D,i} = \sum_{j=1}^n D(i,j)$$

and

$$\mu = N_e - N_v + 1$$

where  $N_e$  and  $N_v$  are the numbers of edges and vertexes in the chemical graph. The only differences is that we used the elements of the  $D'(G)$  matrix instead of the elements of the  $D(G)$  matrix. These four indexes were divided by the number of vertexes in the corresponding chemical graph (number of non-hydrogen atoms).

(5) One more index used for BPNN's training has been derived from Wiener's number:

$$W''(G) = \left( \sum_{i=1}^N \sum_{j=1}^N D'(i,j)(n+1)(I+n) \right) / W(G)$$

where  $D'(i,j)$  is a corresponding element of the  $D'(G)$  matrix,  $N$  is the number of vertexes in the chemical graph,  $n$  is the number of shortest paths between the  $i$ th and  $j$ th vertexes,  $I$  is the number of other shortest paths (between other vertexes) that go via the  $i$ th and  $j$ th vertexes, and  $W(G)$  is the ordinary Wiener number.

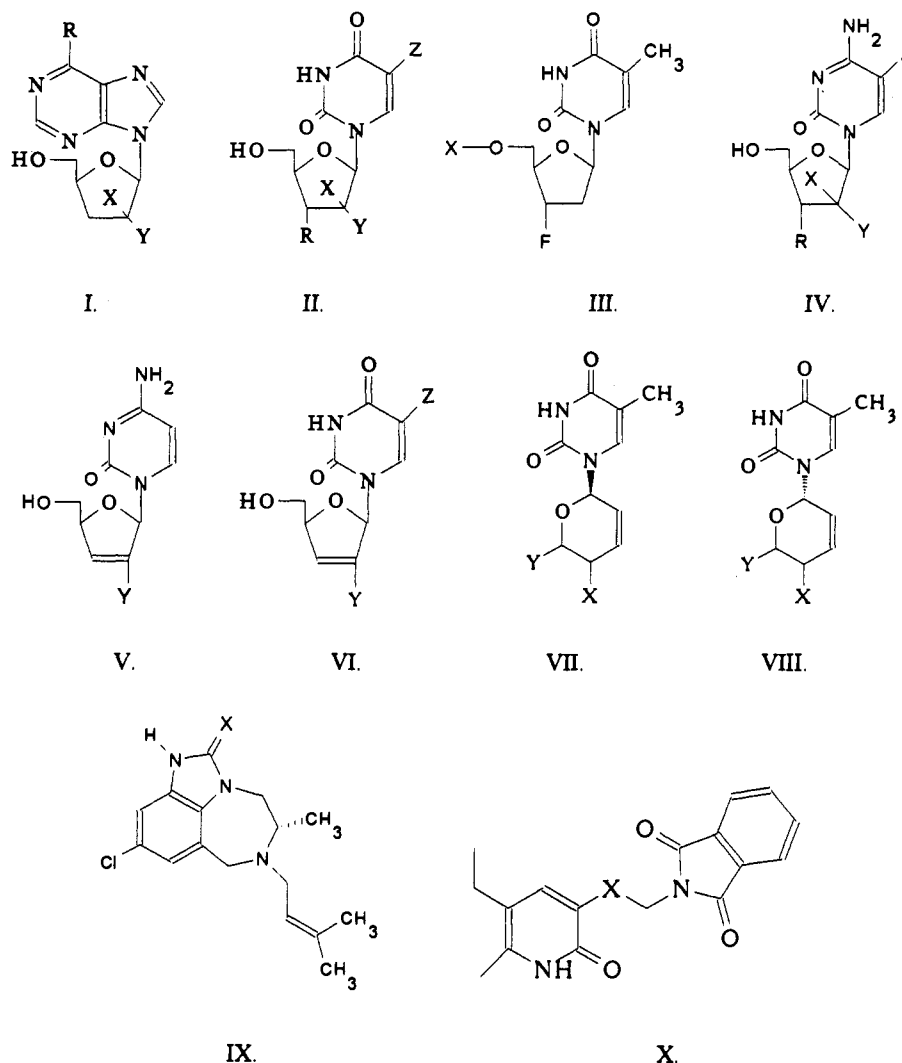
(6) The last index is the so-called code for comparison of structures.<sup>21</sup>

$$M_e(G) = \sum_{i=1}^n v_i^2$$

Here  $v_i$  is the number of vertices of  $i$ th degree, and  $n$  is the maximum vertex degree in a given molecule. The input neurons corresponding to indexes 2 and 6 were pruned as redundant during the network's training.

### Calculation Results

The earliest neural network had 6-10-2 architecture. Six neurons in the input layer correspond to the beginning number of used molecular parameters and two neurons in the output layer to the number of used ranks. The network correctly recognized all compounds from the learning set. This condition was still fulfilled after pruning seven neurons from the hidden layer and two neurons (i.e., input molecular parameters) from the



**Figure 2.** Chemical structures used for inhibitors of HIV-1 RT data base.

**Table 2.** Numbers of Correct Predictions of the Molecules' Activity in 40 Attempts and Final Classification of the Molecules by Neural Networks Ensembles for Different Architectures of the Net<sup>a</sup>

no.	neural net architecture			
	6-10-2	6-3-2	4-10-2	4-3-2
10	38+	37+	40+	40+
12	36+	21 <sup>b</sup> ±	40+	40+
13	36+	29+	27+	22 <sup>b</sup> ±
14	4-	12-	28+	38+
15	5-	17 <sup>b</sup> ±	19 <sup>b</sup> ±	39+
16	3-	14-	34+	40+
19	40+	40+	40+	40+
26	40+	40+	40+	40+
30	40+	40+	40+	40+
35	40+	40+	40+	40+
38	40+	40+	40+	40+
42	40+	40+	40+	40+
43	40+	34+	1-	1-
44	2-	16 <sup>b</sup> ±	39+	40+
number of incorrectly or unclassified molecules	4	5	2	2

<sup>a</sup> The final classification of each molecule was at least at the level  $p < 0.1$  of significance. <sup>b</sup> This molecule cannot be classified at the level  $p < 0.1$  of significance.

input layer. The neurons in the hidden layer were pruned first. Further pruning resulted in an erroneous classification of some molecules from the learning set.

**Table 3.** Cytotoxic Effect of Compounds on MT-4 Cell Culture<sup>a</sup>

investigated compounds	cytotoxic concentration for 50% death of cells (CTC <sub>50</sub> , 10 <sup>-3</sup> M)
AZT (6)	5.3
B1	58.0
B2	46.6
B3	0.078

<sup>a</sup> Starting concentration of MT-4 cells was  $3 \times 10^6$ /mL.

We also tried deleting each parameter from the input set, but it also led to wrong predictions of some molecules from the learning set. The network 4-3-2 was therefore considered as final. The ultimate parameter set is shown in Table 1. In all our simulations, the learning rate was  $\epsilon = 0.1$ , the momentum rate was  $\eta = 0.8$ ,  $\alpha = 0.01-0.1$  (it depends upon the number of neurons in the hidden layer), and the initial weights were in the range  $[-0.5, 0.5]$ .  $\lambda$  and  $\theta_j$  were the same for all neurons:  $\lambda = 1$  and  $\theta_j = 0$ .

We used 40 random starting weight matrices to obtain a statistically significant classification. Each attempt consisted of 10 thousand or less weight updations. In order to avoid local minimums, we began our simulations with the 4-5-2 network with consequent pruning of two neurons. Even in this case, after pruning, the network failed in some tries to correctly recognize all molecules from the learning set. We ignored such

**Table 4.** Effect of Compounds on MT-4 Cells Infected by HIV-1/IIIB

	control (infected MT-4 cells)	measured values					
		tested compounds					
		B1 (M)		B2 (M)		AZT (M)	
		10 <sup>-4</sup>	10 <sup>-5</sup>	10 <sup>-4</sup>	10 <sup>-5</sup>	10 <sup>-4</sup>	10 <sup>-5</sup>
content of viable cells (%)	18.7	37.3	52.1	85.1	83.2	78.6	69.7
no. of infected cells (%)	24.9	19.7	12.6	3.6	2.8	2.9	2.1
antigen p24 content in culture fluid (ng/mL)	1.98	1.92	1.87	0.87	0.77	0.82	0.68
RT activity in culture fluid (%)	100.0	84.3	82.5	20.9	18.7	19.2	17.0

**Table 5.** Performance of Different Methods Compared to That of Neural Networks

	neural networks	k-nearest neighbors	adaptive least squares	linear learning machine
no. of correctly classified molecules in the learning set (30 molecules)	30	23	23	25
no. of correctly classified molecules in the control set (14 molecules)	12	10	11	10

attempts and generated new ones. Table 2 shows the results of the final neural ensembles' classifier. BPNN gave good prediction. Only one molecule, **43**, from the control set was misclassified, and one molecule, **13**, could not be correctly classified. The activities of other molecules were correctly predicted at the level of  $p < 0.1$ . The molecule **13** had the equal probability of being classified as both active and inactive.

To prove the importance of NN pruning for improving the generalization performance of networks, we also utilized 40 calculations using the primary 6 parameter set. The networks had 3 or 10 neurons in the hidden layer. The results are shown in Table 2. The predictions of molecules from the control set were worse in comparison to those obtained for the diminished parameters set. It is interesting to note that the number of neurons in the hidden layer had no significant influence on the generalization performance of BPNN, which significantly increased when the most relevant parameters were used.

The activities of 20 new molecules were predicted. Only molecule **B2** was predicted as active at the level of significance  $p < 0.01$ . The other molecules were predicted at the same level of significance as inactive.

### Biomedical Investigations

Among 20 new substances, only three were chosen for biological testing—**B2**, the substance that should be active according to computer prognosis, and **B1** and **B3**, the nearest analogs of **B2** in chemical structure that should not be active according to the prognosis (Table 1). AZT was used as a reference drug.

Experiments were conducted on MT-4 lymphocyte cultures. For infecting MT-4 cells, the virus-containing fluid of the HIV-1 chronically infected H9/IIIB human T-lymphocyte-passaged cultures was employed. The infection multiplicity in the range of 0.05–0.1 log<sub>10</sub> of EPCD<sub>50</sub>/MT-4 cell was used. The cytopathic effect in MT-4 infected cells according to Reed–Muench method was determined.

**Table 6.** Prediction of Activities of New Molecules by Different Methods

	method			
	neural networks	k-nearest neighbors	adaptive least squares	linear learning machine
<b>B1</b>	inactive	inactive	inactive	inactive
<b>B2</b>	active	active	active	inactive
<b>B3</b>	inactive	active	inactive	inactive

The cytotoxic effect of the compounds was estimated in preliminary experiments by the content of viable cells in uninfected MT-4 cultures through the use of staining in a 0.25% trypan blue solution (Table 3).

**B3** was excluded from further testing because of its too high cytotoxicity. That is why we used only **B1** and **B2** with reference substance AZT in experiments on infected cells.

Infected MT-4 cells were cultivated in the presence of tested compounds at concentrations of 10<sup>-4</sup> and 10<sup>-5</sup> M for 7 days. The cytotoxic effect of the virus on the infected MT-4 was estimated by the same method with 0.25% trypan blue solution. The number of infected cells, expressing viral antigens on superficial membranes, was determined by an indirect immunofluorescence assay using anti-HIV-1 antibodies.<sup>25</sup> The level of virus production in cultures was estimated by the content of viral antigen p24 in a culture fluid using an immunoenzymatic assay.<sup>26</sup> The level of virus production was also estimated by RT activity in the culture fluid.<sup>27</sup>

Data obtained (Table 4) show that the computer prognosis is in full compliance with the results of biological testing. Substance **B2** is approximately as active against HIV-1 RT as AZT but 10 times less toxic for cells than AZT.

### Conclusion

The main advantage of neural networks is their nonlinear mapping. It is known that three-layer neural networks (and also higher layer networks) can approximate any multidimensional function with given accuracy and can exactly implement an arbitrary finite training set (a global existence theorem was formulated by A. N. Kolmogorov<sup>28</sup> and a possible application to neural networks was suggested by R. Hecht-Nielsen<sup>29</sup>). Biological phenomena are considered nonlinear by nature. Therefore, the contribution of physicochemical and substructural parameters to biological activity can be nonlinear, and this property of network mapping is very important for SAR and QSAR studies. However, the network can easily overfit. It will result in incorrect, biased predictions of new molecules. There is a number of methods that can improve BPN's generalization. Two of them, namely NN ensembles and network pruning,

were successfully used in this work. The others are training with a noise, stopping at a sensible level of  $E_{gl}$  (judged either manually or by cross-validation), and distributed bottleneck, etc. We are investigating now the suitability of their use in structure-activity relationship problems.

Estimating molecular parameters during learning can be easily incorporated into back-propagation training and seems to be a useful tool for the researcher. However, it cannot be used to draw a few significant parameters from a large parameter set because of a very large amount of calculations. Faster simple methods (i.e. based on cluster analysis) should be used for preprocessing a large data set. Final evaluation and choosing of the most significant parameters can be done by a network itself.

The performance of neural networks has been compared to the performance of several other methods, namely adaptive least squares,<sup>30</sup>  $k$ -nearest neighbors,<sup>31</sup> and linear learning machine<sup>32</sup> (Tables 5 and 6). Although the activity of **B2** is usually predicted correctly, the performance of neural networks on learning and control sets of molecules is better.

## References

- (1) *Parallel Distributed Processing Exploration in Microstructure of Cognition*; Rumelhart, D. E., McClelland, J. L., Eds.; MIT Press: Cambridge, MA, 1986; Vols. 1 and 2.
- (2) Andrea, T. A.; Kalayeh, H. Applications of Neural Networks in Quantitative Structure-Activity Relationships of Dihydrofolate Reductase Inhibitors. *J. Med. Chem.* **1990**, *33*, 2583-2590.
- (3) Livingstone, D. J.; Manallack, D. T. Statistics using neural networks: chance effects. *J. Med. Chem.* **1993**, *36*, 1295-1297.
- (4) Tetko, I. V.; Luik, A. I.; Poda, G. I. Application of Neural Networks in structure-Activity Relationships of a Small Number of Molecules. *J. Med. Chem.* **1993**, *36*, 811-814.
- (5) Baum, E. B.; Haussler, D. What size net gives valid generalization? *Neural Computat.* **1989**, *1*, 151-160.
- (6) Denker, J.; Schwartz, D.; Witner, B.; Solla, S.; Hopfield, J.; Howard, R.; Jacke, L. Automatic learning, rule extraction and generalization. *Complex Systems* **1987**, *1*, 877-922.
- (7) Sietsma, J.; Dow, R. J. F. Creating Artificial Neural Networks That Generalize. *Neural Networks* **1991**, *4*, 67-79.
- (8) Kruschke, J. K.; Movellan, J. R. Benefits of Gain: Speeded Learning and Minimal Hidden Layers in Back-Propagation Networks. *IEEE Trans. Systems Man Cybernet.* **1991**, *21*, 273-280.
- (9) Hirose, Y.; Yamashita, K.; Hijiya, S. Back-Propagation Algorithm Which Varies the Number of Hidden Units. *Neural Networks* **1991**, *4*, 61-66.
- (10) Tetko, I. V.; Luik, A. I. A node pruning algorithm for feed-forward neural networks. *J. Intell. Control. Neurocomput. Fuzzy Logic* **1993**, *1*, in press.
- (11) Hansen, L. K.; Salamon, P. Neural Networks Esembles. *IEEE Trans. Pattern Anal. Machine Intell.* **1990**, *12*, 993-1001.
- (12) Baba, M.; Pauwels, R.; Herdewijn, P.; Glercq, E. D.; Desmyter, J.; Vandeputte, M. Both 2',3'-dideoxythymidine and its 2',3'-unsaturated Derivative (2',3'-deoxythimidine) are Potent and Selective Inhibitors of Human Immunodeficiency Virus Replication in vitro. *Biochem. Biophys. Res. Commun.* **1987**, *142*, 128-134.
- (13) Martin, J. A.; Bushnell, D. J.; Duncan, I. B.; Dunsdon, S. J.; Hall, M. J.; Machin, D. J.; Merrett, J. H.; Parkes, K. E. B.; Roberts, N. A.; Thomas, G. J.; Galpin, S. A.; Kinchington, D. Synthesis and Antiviral Activity of Monofluoro and Difluoro Analogs of Pyrimidine Deoxyribonucleosides against Human Immunodeficiency Virus (HIV-1). *J. Med. Chem.* **1990**, *33*, 2137-2145.
- (14) Sterzycki, R. Z.; Ghazzouli, I.; Brankovan, V.; Martin, J. C.; Mansuri, M. M. Synthesis and Anti-HIV Activity of Several 2'-Fluoro-Containing Pyrimidine Nucleosides. *J. Med. Chem.* **1990**, *33*, 2150-2157.
- (15) Bessodes, M.; Egron, M.-J.; Filippi, J.; Antonakis, K. Synthesis of Unsaturated 4'-Azido Pyranosyl Thymine as Potent Antiviral and anti-HIV Agents. *J. Chem. Soc., Perkin. Trans 1* **1990**, 3035-3041.
- (16) Marquez, V. E.; Christopher, K.-H. T.; Mitsuya, H.; Aoki, S.; Kelley, J. A.; Ford, H.; Roth, J. S.; Broder, S.; Johns, D. G.; Driscoll, J. S. Acid-Stable 2'-Fluoro Purine Dideoxynucleosides as Active Agents against HIV. *J. Med. Chem.* **1990**, *33*, 978-981.
- (17) Sarri, W. S.; Wai, J. S.; Fisher, T. E.; Thomas, C. M.; Hoffman, J. M.; Rooney, C. S.; Smith, A. M.; Jones, J. H.; Bamberger, D. L.; Goldman, M. E.; O'Brien, J. A.; Numberg, J. H.; Quintero, J. C.; Schleif, W. A.; Ermini, E. A.; Anderson, P. S. Synthesis and Evaluation of 2-Pyridinone Derivatives as HIV-1-Specific Reverse Transcriptase Inhibitors. 2. Analogues of 3-Aminopyridin-2(1H)-one. *J. Med. Chem.* **1992**, *35*, 3792-3802.
- (18) Ho, C. Y.; Kukla, M. J. Synthesis of the Pyrimidine Analog of 4,5,6,7-Tetrahydroimidazol[4,5,1-jk][1,4]Benzodiazepin-2(1H) ONE (TIBO) Potential for HIV-1 Inhibition. *Bioorg. Med. Chem. Lett.* **1991**, *1*, 531-534.
- (19) McGuigan, Ch.; Jones, B. C. N. M.; Devine, K. G.; Nicholls, S. R.; O'Connor, T. J.; Kinchington, D. Synthesis and Evaluation of some Novel Phosphoramidated Derivatives of 3'-azido 3'-deoxythymidine (AZT) as Anti-HIV Compounds. *Bioorg. Med. Chem. Lett.* **1991**, *1*, 729.
- (20) Visnevskii, S. G.; Pirozhenko, V. V.; Chentsova, N. P.; Antonenko, S. V.; Barbasheva, E. V.; Grin', E. V.; Lyul'chik, M. G.; Sorochinskii, A. E.; Remennikov, G. Ya.; Luik, A. I.; Kukhar, V. P. Synthesis and Anti-viral Activity of 2',3'-dihydroxypropyl derivatives of 5-nitro-2,5- and -1,6-dihidropyrimidines. *Khim.-Farm. Zhurn.*, in press.
- (21) Gutman, I.; Randic, M. Algebraic Characterization of Skeletal Branch. *Chem. Phys. Lett.* **1977**, *47*, 15.
- (22) Kier, L. *Molecular Connectivity in Chemistry and Drug Research*; Academic Press: New York, 1976.
- (23) Barysz, M.; Jashari, G.; Lall, R.; Srivastava, V.; Trinajstic, N. Distance matrix for molecules containing heteroatoms. In *Chemical Applications of Topology and Graph Theory*; King, R., Eds.; Elsevier: Amsterdam, Oxford, New York, Tokyo, 1983; pp 222-234.
- (24) Balaban, A. Topological indexes based on topological distances in molecular graphs. *Pure Appl. Chem.* **1983**, *55*, 199-206.
- (25) Jackson, J. B.; Balfour, H. H. Practical diagnostic testing for human immunodeficiency virus. *Clin. Microbiol. Rev.* **1988**, *1*, 124-138.
- (26) Tremblay, M.; Sullivan, A.; Roske, R. New CD<sup>+</sup> cell line susceptible to infection by HIV-1. *Med. Virol.* **1989**, *28*, 243-249.
- (27) Yoshiyama, H.; Kobayashi, S.; Tanade, A. Determination of human immunodeficiency virus (HIV) in cultures in lymphocytes from HIV-seropositive persons with special reference to an enzyme immunoassay. *J. Infect.* **1989**, *19*, 143-151.
- (28) Kolmogorov, A. N. On the representations of continuous functions of many variables by superpositions of continuous functions of one variable and addition. *Dokl. Akad. Nauk USSR* **1957**, *114*, 953-956.
- (29) Hecht-Nielsen, R. Kolmogorov's mapping neural network existence theorem. *Proceedings of the International Conference on Neural Networks*; IEEE Press: New York, 1987; pp 11-14.
- (30) Moriguchi, I.; Komatsu, K.; Matsushita, Y. Adaptive least-squares method applied to structure-activity correlation of hypotensive N-alkyl-N''-cyano-N'-pyridylguanidines. *J. Med. Chem.* **1980**, *23*, 20-26.
- (31) Tetko, I. V.; Tanchuk, V. Yu.; Luik, A. I. Application of an evolutionary algorithm to the structure activity relationship. *Proceedings of the Third Annual Conference on Evolutionary Programming*; Sebald, A. V., Fogel, L. J., Eds.; World Scientific: River Edge, NJ, 1994; pp 109-119.
- (32) Saaki, S.; Abe, Y.; Takahashi, Y.; Takayama, T.; Miyashita, Y. *Introduction to pattern recognition for chemists*; Tokyo Kagaku Dojin: Tokyo, 1984.