Scientific paper

Linear vs. Non-linear Modelling. Case study: modelling of binding affinity of inhibitors to Trypsin

Jure Zupan and Špela Župerl

Laboratory of Chemometrics National Institute of Chemistry, Ljubljana, Hajdrihova 19, SI-1000 Ljubljana, Slovenia

** Corresponding author: E-mail: jure.zupan@ki.si*

Received: 08-04-2011

Dedicated to Professor Dušan Hadži on the occasion of his 90th birthday

Abstract

On the set of 53 trypsin inhibitors the affinity to the covalent bound ligands is modeled using linear (MLR) and non-linear (ANN) methods. Each compound is represented by 343 chemical descriptors. The hypothesis was that linear models are not sufficiently flexible to yield the best model, because in MLR (multiple regression analysis) the number of variables (descriptors) is limited by the number of objects in the training set. On the other hand the CP-ANN (counter-propagation-artificial neural network) is not limited by this restriction and can thus involve larger number of variables than there are compounds in the training set. Both methods are applied on the same division of 53 compounds on the training, test, and validation sets. In a systematic GA (genetic algorithm) search the MLR models containing all possible forms of linear polynomials, i.e., from 3 to 25 variables were scanned and no better model that one obtained by the CP-ANN model was found.

Keywords: Genetic algorithm (GA) optimization, multiple linear regression (MLR) modeling, counter-propagation artificial neural networks (CP-ANN) modeling, trypsin complexes, quantitative structure activity relationship (QSAR)

1. Historical Remark

Quite a long time ago (in fall of 1972) Professor Dužan Hadži suggested to one of us (JZ) to participate at the Noordwijkerhout *Advanced Study Institute on Computer Representation and Manipulation of Chemical Data*. Even after almost forty years this conference is still regarded as one of the milestones in the field of Computerized Chemical Information Science. Professor Hadži's idea was to introduce this branch of chemistry to Slovenian science community. At that time in his Laboratory there was already a very strong group (lead by the late Professor Andrej Ažman) working with computers of that time on Quantum Chemistry problems. Nevertheless, Professor Hadži has envisaged that in the future the Quantum Chemistry will not be the only field in chemistry where the computers will play a significant role, so *via* our group of Chemometrics which originated from the 1973 in his Laboratory, he promoted many long-term projects in different directions of computerized chemical information science. Now, besides several groups working on a variety of quantum chemical problems from *ab intio* calculations to protein folding simulations, many other areas of computer research in chemistry such as combined spectral information systems, pattern recognition and artificial intelligence, chemometrics, QSAR, artificial neural networks, and many other studies have been developed and well established in Slovenian chemical community. With years the motto supported by professor Hadži *'do not regard computers as number crunching machines, but use them as the experimental equipment'* has attracted scores of excellent and capable scientist which have produced many excellent research results. All of us working in these areas would like at the occasion of his 90th birthday to compliment his vision of that time.

2. Introduction

The basic equation of the Hansch approach[1] to the Quantitative Structure-Activity Relationship[2] (QSAR) is the regression line that describes N activities $\{A_i\}_{i = 1,\ldots N}$

of a set of N compounds $\{ X_i \}_{i = 1,\ldots N}$ each represented by p variables or descriptors; $X_i = (x_{i1}, x_{i2}, \ldots x_{ip})$ with a system of N linear equations containing p variables x_{ij}:

$$A_i = b_0 + \sum_{j=1}^{p} b_j x_{ij} ; \quad i = 1,\ldots N \tag{1}$$

The standard solution of the system of N linear equations /1/ yields $p + 1$ coefficients b_j, $j = 0,1,\ldots p$. The underlying assumption is that the applied molecular or structural descriptors actually influence the activity A in question. The magnitudes and signs of the coefficients b_j, $j = 0,1,\ldots p$ reflect the intensity and direction of the influence of each variable or descriptor on the activity A. In general there are two goals within the QSAR research. The first one tries to predict the activities of new compounds using Multiple Linear Regression (MLR) model /1/, i.e., the same set of variables x_j, $j = 1,\ldots p$ and the same set of coefficients b_j, $j = 0,1\ldots p$. The other goal is directed toward the selection of the molecular descriptors or variables that influence the known activities. In a complex QSAR research most often both goals must be pursued. Regarding the fact that nowadays thousands of structural descriptors can be easily generated for each molecule the second task, i.e., a selection of most appropriate descriptors is carried out first. It is understandable that a MLR system /1/ cannot have N very large (in order of hundreds or even thousands) because this would requires at least $N + 1$ compounds with known activities to be available in order to solve the system. Mostly the studies of QSAR are carried out with hundred or somewhat less compounds what heavily limits the number of molecular descriptors.

The limitation of the number of variables for using MLR is the reason that other methods for modeling were tried, for example Artificial Neural Network (ANN) methods,[3] specifically, Counter-propagation ANN (CP-ANN). Due to its so-called 'half-unsupervised character' the advantage of the CP-ANN method is that it is not restricted to the discussed limitation of variables. However, the question remains whether the nonlinear methods are actually better suited for this kind of studies compared to the standard linear ones. In other words, is there actually a need for employing the ANN methods in the QSAR studies. Therefore, our goal was to make an extensive study within the complete space of all available MLR polynomials (linear in the respect to the variables as required by the classical QSAR equation /1/) for a given set of data for which an optimized nonlinear modeling already exists[4] and check out whether a better MLR can possibly be found or not.

3. Methods

Genetic Algorithm (GA) is well known optimization procedure for the selection of variables from larger set. However, for our study in which we try systematically investigate all possible polynomial configurations (i.e., taking into account all possible forms of linear polynomials from $p = 3$ to $p = 25$) the standard GA[5] procedure is not best suited. Therefore, the so called *permutation* representation[6] of chromosomes (or order-based representation) instead of the bit-wise or sequentially ordered one has been applied. Accordingly, the crossover and the mutation functions have been adjusted to the used permutation representation of chromosomes.

In the standard chromosome representation for the selection of p out of the larger set of P descriptors for making the best model, each chromosome representing one possible solution consists of a string of P zeros and ones /2/, with p ones representing the presence of the molecular descriptor associated with a given position in the string of length P:

Chromosome = (0,0,0,1,0,0,1,0,0,0,1,0,0,0,
0,1,0,0, 0, … 0,0,0,1,0) (2)
P positions, p ones

On the other hand in the *permutation* representation of chromosomes each chromosome is represented by a *permuted sequence* of P numbers from 1 to P, with the first p numbers yielding the possible selection of p molecular descriptors used in the MLR modeling /3/:

Chromosome = (4,234,32,8,224,330,67,25,
132,P-2, …, P, … P-1,65) (3)
P numbers 1 to P

The crossover is performed in exactly the same way as in the standard representation with an additional check that assures removing of the duplicates of any gene number in the upper parts of both descendant chromosomes which may arise through the interchange of lower halves of parents. The mutation is made by switching the numbers at two randomly selected gene positions in the chromosome. By taking into account always the same first p numbers in the chromosome one can assure that in a given GA optimization run there are always exactly the same number of molecular descriptors in the MLR system /1/. In this way one can focus a special study on the optimization of a single type of polynomials (polynomials with a pre-specified p) at the time. In the presented work 22 such studies (for $p = 3$ to 25) were made for each single polynomial type p. In this way we were sure that polynomials with different number of coefficients p were covered equally and adequately.

The fitness function of each chromosome can be either the correlation coefficient between the activities predicted by the model on the test set compounds $\{y^{\text{test}}_i\}^{\text{model}}$ and the experimental activities $\{y^{\text{test}}_i\}^{\text{experiment}}$ or the Root-Squared-Error (RSE) between the same sets of activity data y_i be in the training, test, or validation set, respecti-

vely, (4):

$$RSE = \sqrt{\frac{\sum_{i=1}^{n} (y_i^{experiment} - y_i^{model})^2}{n}} \quad (4)$$

The evaluation of one pool of n_c chromosomes thus requires building up of n_c MLR multiple linear regression systems[7] /1/ always with p variables:

$$Y^{training} = \|X^{training}\| B^T \quad (5)$$

Having the solutions for n_c sets of coefficients $B = (b_0, b_1, \ldots b_p)$, on the training compounds in the form:

$$B^T = ((\|X^{training}\|^T \times \| X^{training}\|)^{-1} \times \| X^{training}\|^T) \times Y^{training} \quad (6)$$

the fitness value of the particular chromosome is calculated on the predictions Y^{test}

$$Y^{test} = \|X^{test}\| B^T \quad (7)$$

comparing them with the experimental values $Y^{test,experiment}$. In each pool all chromosomes are ordered according to their fitness function and a new pool is generated using the standard GA procedures. In our studies a minimum of 15,000 pools each having 50 chromosomes (750,000 MLR calculations) were inspected for each polynomial type, i.e., for each p. Very often different GA parameters as mutation rate, elitism (on/off), or different numbers of pools were tried to find the optimal solution. This means that on the average more than one million MLR calculations were made for each p. There is no specific excluding procedure for descriptors. The descriptors are initially chosen at random from the 343 possible ones – with no bias or threshold on any of them. Further on, during the GA procedure, the genes are 'moving' from one generation to another strictly *via* the rules of crossing, mutations and selection of the best based on the roulette wheel parent selection[6].

For the non-linear modeling the Counter-propagation artificial neural network (CP-ANN) methods was used. CP-ANN model is an upgrade of Kohonen mapping[3]. The stand alone Kohonen network only clusters the input signals (sets of descriptors), $\|X^{training}\|$, into 2-dimensional plane of neurons. In the CP-ANN model an extra layer of neurons (Grossberg layer) is added to the Kohonen layer, having exactly the same number and layout of neurons as the Kohonen one. To the Grossberg layer the responses (activities) $Y^{training}$ are input. Therefore, the self-organization in the Kohonen layer produces the corresponding rearrangement and 'smoothing' of responses $Y^{training}$ over all cells of the Grossberg layer. By input of an 'unknown' (test or validation signal (i.e., a set of descriptors X_i^{test} or X_i^{valid}) into the Kohonen layer and by recording the position of the most excited neuron in this layer, one can retrieve at the same position in the Grossberg layer the corresponding activity, $y^{ANN\text{-}model}$.

In order to obtain the optimal CP-ANN model for prediction of inhibition constants the following network parameters were varied; the number of neurons in the ANN from 5 × 5 to 9 × 9, the number of learning epochs from 1 to 1000, maximal learning rate from 0.1 to 0.9 and minimal learning rate from 0.01 to 0.1. The GA was coupled with CP-ANN to reduce the number of descriptors included in the models. A population of 100 chromosomes evolving in 600 generations was considered in each combination of different network and GA parameters.

3. 1. The Data

In the present investigation[4,8] we have used a master dataset of 53 trypsin inhibitors with known binding affinities to the ligands. The structure of each compound $X_i = (1, x_{i1}, x_{i2}, \ldots x_{ij}, \ldots x_{i343})$, $j = 1,\ldots 343$ is described by 343 molecular descriptors x_{ij} and known activity or dependent variable y_i, $i = 1,\ldots 53$. Activity y_i of the compound X_i is defined as its binding affinity to trypsin. The molecular descriptors were obtained using CODESSA program[9,10]. The initial number of more than thousand descriptors provided by the CODESSA program was reduced by employing several simple checks for complete identity, for zero descriptors, for descriptors having very low variance (less than 0.001), and/or for not applicable descriptors to sets compounds, etc. At the end of this checks 343 descriptors remained. Hence, all GA optimization runs started by the compounds described with 343 descriptors, or 'genes', each having unique ID in the range from 1 to 343. As mentioned before, three data matrices: training, test, and validation matrices: $\|X^{training}\|$, $\|X^{test}\|$, and $\|X^{valid}\|$ with dimensions (26 × 344), (15 × 344) and (12 × 344), respectively, were constructed. The first column of each data matrix $\|X\|$ is equal to 1. To each data matrix $\|X\|$ a corresponding activity vector $Y^{training}$, Y^{test} or Y^{valid} with 26, 15, and 12 activity values, respectively, is associated. The division of the master on the three sub-set using the Kohonen neural network[7] has been already described by one of us (ŠŽ)[8]. The Kohonen neural network with the input of all 53 compounds has provided a self-organized top map of positions of neurons excited by each compound. From 53 points (positions of neurons excited by 53 compounds) on the top map, first, the validation and then the test set were extracted. The remaining compounds were used as the training set. The criterion for the selection of the validation and test set was the requirement that the compounds in each set cover the space of the top map as evenly as possible. Additionally, we try to balance the numbers of compounds in each set in such a way that the ratio of the training/(validation + test set) was 1:1 and then to keep approximately the same ratio for the validation/test set. The result is the division of 12, 15, and 26 compounds in the validation, test, and training set, respectively.

4. Results and Discussion

The cumulative results of the study are shown in Figure 1. The performance of the best models' predictions were found for all p (p = 3 to 25). As expected, the RSE values for models obtained for the training sets is decreasing towards zero as the number of polynomial coefficients p approaches the number of compounds in the training set. With the actual data set containing relatively small number of compounds (26) in all three sets we have first to resolve the issue of the best fitness function *ff*. When the preliminary GA optimizations were made it turns out that neither RSE (eq. /4/) nor the correlation coefficient r describes the fitness of the optimized solution (model) best. If testing only RSE the correlation coefficients of the predicted values of the test set were low indeed, on the other hand, the models yielding higher correlation coefficients have RSE values in the order of activities or even more. Then the compromise was made taking into account the ratio between both values as a fitness function *ff*:

$$ff = \frac{RSE}{r^2} \tag{8}$$

and again, using equation /8/ as the fitness function, the entire space of various MLR polynomials (p = 3 to 25) was scanned in the search for the best GA parameters (crossover rate, mutation rate, size of the pool, number of iterations, survival rate). After the inspection of the results we have decided to run the same three data sets (training, test and validation) for all polynomials with the same set of GA parameters (elitism included, 50 chromosomes in the pool, for 5000 generations, mutation rate 0.05, crossover rate 0.98, and survival rate dependent of 5 best chromosomes). Additionally, it turns out that for the entire polynomial space (all sizes of polynomials) the combined fitness function *ff* (eq. /8/) wasn't as good as it was supposed to be. For polynomials having small number of coefficients ($p \leq 10$) the fitness function /8/ was acceptable, while for longer polynomials ($p > 10$) it starts favoring the solutions with higher correlation coefficients over the ones giving smaller RSE. In order to avoid this problem, we have decided that RSE (eq. /4/) describes the fitness results over the whole range of polynomials better than any other formula we have tried. This, of course, is not to say that no better fitness function could be found. Hence, all GA optimizations were run again and using the RSE value obtained on the *test* set as a fitness function.

Figure 1 shows the resulting best RSE values of the described final GA runs (with final agreed parameters) for all polynomials. As expected, the RSE values between the model prediction and the experimental activities from the training sets (26 compounds) decrease with the increasing number of coefficients p. Clearly, the RSE value is very close to zero for the optimized polynomial with 25 coefficients. On the first sight it may seem strange that the optimized RSE values obtained with the training set are higher than that for the test set. However, the entire optimization procedure is not made in a single GA run, but it requires to find the MLR model (using MLR procedure eq. /4/) obtained on the *training* set that yields the best RSE on the *test* set (eq. /5/) using the coefficients obtained in /4/. Optimized RSE values obtained with the *training* set only could of course be lower if the optimization would not be restricted by the condition that the GA procedure's fitness function *ff* is taken on the *test* set. From the lower curve in Figure 1 (RSE values of the test set) it can be seen that the best final RSE^{test} values using the *test* set (15 compounds) for a given optimization, with p 'genes' turned 'on' in the chromosomes of 343 genes, has a broad minimum around p = 10.

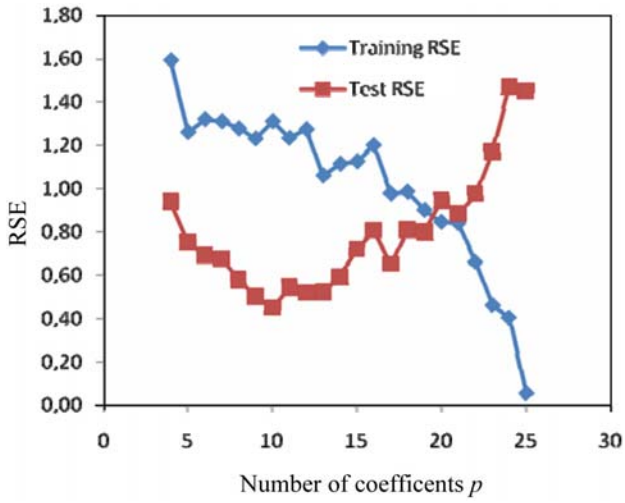
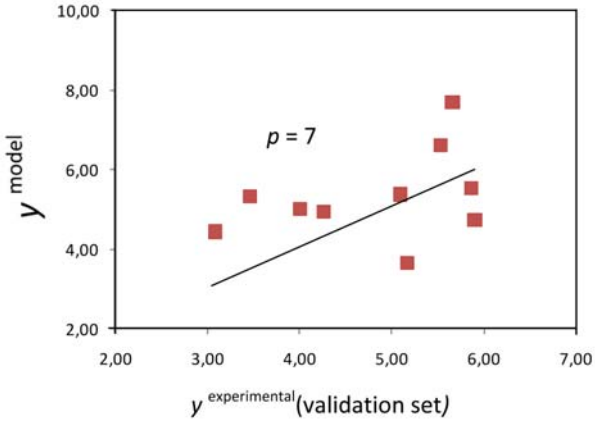


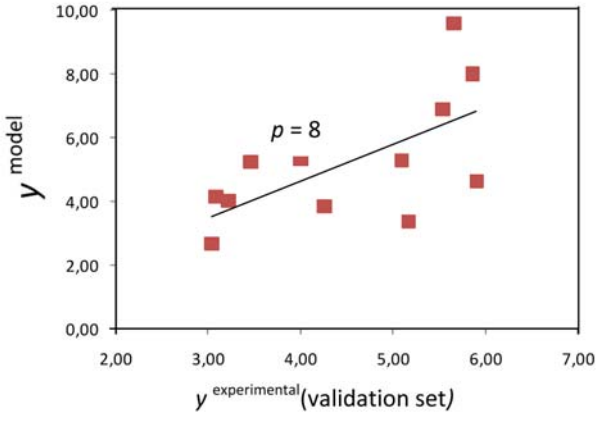
Figure 1. The best (optimized by the GA procedure) RSE values (equation /4/) of the training and test set for different forms of MLR polynomials (p = 4 to 25).

Therefore, in the region from p = 1 to p = 14 all optimized solutions (MLR models obtained on the training set) have been tried. The best three sets of prediction results (predicted activities) on the validation set y_i^{valid}, i = 1,… 12, obtained by coefficients B (eq. /6/) of the best models are shown in Figure 2. The shown predictions calculated by the three best models are the very best results that we could obtain during our extensive GA optimization. From the practical point of view the prediction results are far from being good and reliable. Considering only the correlation and RSE^{valid} one would say that the MLR model for p = 8 (r = 0.62 and RSE^{valid} = 1.68, middle part of Figure 2) is the worst one. On the other hand, considering the distribution of the predicted activities of model p = 8 and its individual agreements with the experimental values in comparison with the other two models (p = 7, p = 9) has brought us to judge this model as the best

one. This example shows how difficult it is to find a good fitness function for such types of optimization.

On the side of non-linear modeling the optimal CP-ANN model[4,8] yielding the RSE^{valid} = 0.71 was achieved with the ANN dimension of 6 × 6 neurons, 200 learning epochs, and with maximal and minimal learning rates of 0.5 and 0.1, respectively. Comparing the predictions by the best CP-ANN procedure (Figure 3) with our data (Figure 2) one can see a clear difference. First, it is evident that both, the correlation coefficient and RSE (r = 0.78, RSE^{valid} = 0.71) values of the CP-ANN model are better than the ones obtained by the MLR models. However, as a defense to the MLR models it should be mentioned that the predicted values in the CP-ANN model are clustered into two groups each of which does not differentiate well the 'fine-structure' of the activities while in the MLR models this is not the case. What the CP-ANN method does better than MLR is the separation of compounds into those with low and the ones with high activities what is the most important and valuable feature.





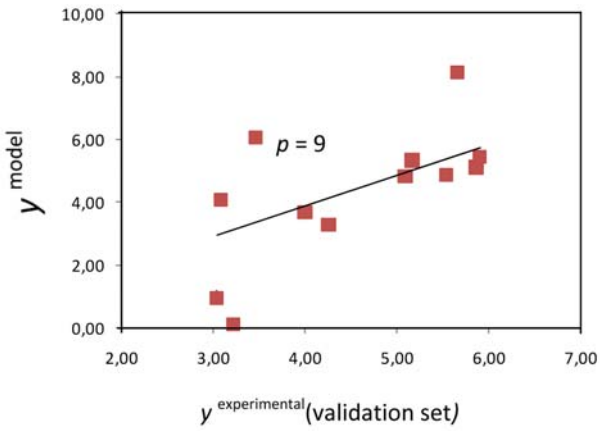


Figure 2. Three best models according to the predictions obtained on the validation set of 12 compounds never used in the training or test procedure. The three best models were obtained with GA procedure with the training and test compounds on the polynomials with seven, eight, and nine coefficients (p = 7, 8, and 9). They have the r / RSE^{valid} values of 0.65 / 1.55; 0.62 / 1.68; and 0.67 / 1.52, respectively.

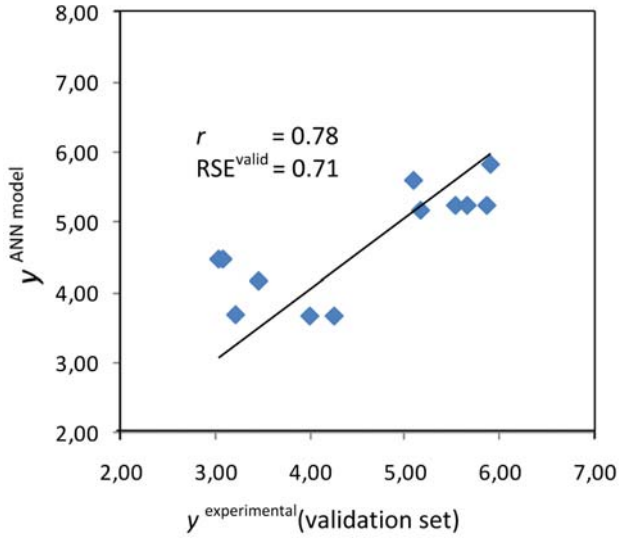


Figure 3. The best CP-ANN model obtained on the same training, test and validation sets of 53 compounds as reported in reference[8].

The present study was made primarily to find out the limits of the MLR method in the view of a sparse number of available data. What the problem of the selection of descriptors is concerned, it is difficult to compare a method (MLR) in which the number of descriptors is limited by the size of the training set with a non-linear method (CP-ANN) which does not have this restriction. In Table 1 all 97 molecular descriptors selected by the best CP-ANN model (Fig. 3) and 7 molecular descriptors selected by the best MLR model (Fig.2, p = 8) are shown. There are four descriptors of the best MLR model that are found among the ones selected by the CP-ANN model what shows the reliability of both methods.

5. Conclusion

The present study was made with the intention to explore the limits of the MLR methods for a given case and

Table 1. Molecular descriptors selected by the CP-ANN model (97 – upper part) and by the MLR model (3 – lower part). The descriptors selected by both methods (4) are shown in bold in the upper part.

ID No.	Name of descriptor[9]	ID No.	Name of descriptor[9]
1	Number of atoms	142	HOMO-1 energy
2	Number of C atoms	143	HOMO energy
7	Number of bonds	146	HOMO – LUMO energy gap
14	Relative number of single bonds	149	Avg nucleoph. react. index for a N atom
17	Number of aromatic bonds	152	Avg nucleoph. react. index for a C atom
23	Relative number of aromatic bonds	153	Min electroph. react. index for a N atom
27	Wiener index	183	WPSA-1 (PPSA1*TMSA/1000)
30	Randic index (order 2)	201	RNCG (QMNEG/QTMINUS)
31	**Randic index (order 3)**	204	FHDSA Fractional HDSA (HDSA/TMSA)
33	Kier&Hall index (order 1)	206	FHASA Fractional HASA (HASA/TMSA)
37	Kier shape index (order 2)	207	HBSA H-bonding surface area
44	Average Complementary Info. Cont. (order 0)	209	HDCA H-donors charged surface area
48	Average Info. cont. (order 1)	211	HACA H-acceptors charged surface area
52	Average Complementary Info. cont. (order 1)	218	HA dependent HDSA-1
58	Average Structural Info. cont. (order 2)	221	HA dependent HDSA-2/TMSA
61	Complementary Info. cont. (order 2)	234	HACA-1/TMSA
62	Average Bonding Info. cont. (order 2)	240	Max SIGMA-SIGMA bond order
63	Bonding Info. cont. (order 2)	242	Max PI-PI bond order
65	Moment of inertia A	243	Max bonding contribution of a MO
66	Moment of inertia B	246	Max valency of a N atom
68	XY Shadow	247	Avg valency of a N atom
72	ZX Shadow	248	Min (>0.1) bond order of a N atom
73	ZX Shadow / ZX Rectangle	249	Max bond order of a N atom
74	Molecular volume	250	Avg bond order of a N atom
76	Molecular surface area	253	Avg valency of a C atom
78	Min partial charge for a C atom	257	Min valency of a H atom
80	Min partial charge for a N atom	266	Max e-n attraction for a N atom
82	Min partial charge for a H atom	267	Min atomic state energy for a N atom
85	Polarity parameter (Qmax-Qmin)	268	Max atomic state energy for a N atom
86	Polarity parameter / square distance	269	Min e-e repulsion for a C atom
89	TMSA Total molecular surface area	271	Min e-n attraction for a C atom
90	PPSA-1 Partial positive surface area	273	Min atomic state energy for a C atom
95	WPSA-1 (PPSA1*TMSA/1000)	276	Max e-e repulsion for a H atom
97	PPSA-2 Total charge weighted PPSA	283	Min exchange energy for a C-C bond
98	PNSA-2 Total charge weighted PNSA	285	Min e-e repulsion for a C-C bond
100	FPSA-2 Fractional PPSA (PPSA-2/TMSA)	287	Min e-n attraction for a C-C bond
106	DPSA-3 (PPSA3-PNSA3)	292	Max coulombic interaction for a C-C bond
107	FPSA-3 Fractional PPSA (PPSA-3/TMSA)	296	Max resonance energy for a C-N bond
108	FNSA-3 Fractional PNSA (PNSA-3/TMSA)	**297**	**Min exchange energy for a C-N bond**
112	RPC-SA (SAMPOS*RPCG)	301	Min e-n attraction for a C-N bond
113	RNCG e (QMNEG/QTMINUS)	302	Max e-n attraction for a C-N bond
114	RNCS Relative negative charged SA	304	Max n-n repulsion for a C-N bond
115	**min(No.of HA, No. of HD)**	305	Min coulombic interaction for a C-N bond
119	HA dependent HDSA-1/TMSA	306	Max coulombic interaction for a C-N bond
122	HA dependent HDSA-2/SQRT(TMSA)	**307**	**Min total interaction for a C-N bond**
133	HACA-1	310	Max resonance energy for a C-H bond
138	Final heat of formation	311	Min exchange energy for a C-H bond
139	Final heat of formation / # of atoms	314	Max e-e repulsion for a C-H bond
140	No. of occupied electronic levels		
38	Kier shepe index (order 3)	324	Tot. molec. 1-center E-N attract./No.of atoms
168	Min net atomic charge for a C atom		

to show the justification of the use of ANN methods in this and similar QSAR studies. Due to the enormous size of the search space of possible distributions of 343 molecular descriptors into different types of polynomials from size $p = 3$ to $p = 25$, it is evident that no one can assure that the solution found by the MLR is the best possible one. However, regarding the slow improvement rates during the GA procedures that we have followed in all types of

polynomial optimizations one can be pretty sure that the solution obtained using CP-ANN method is at least as good if not better in terms of prediction quality and robustness than the best MLR model that could be found.

6. References

1. C. Hansch, Act. Chem. Res. **1969**, 2, 232–239.
2. H. Kubinyi, *QSAR: Hansch Analysis and Related Approaches in Book Series: Methods and Principles in Medicinal Chemistry*, VCH, Weinheim, **1993**; Published Online: **2008**. http://onlinelibrary.wiley.com/book/10.1002/9783527616824.
3. J. Zupan, J. Gasteiger, *Neural Networks and Drug Design,* Wiley-VCH, Weinheim, **1999**.
4. Š. Župerl, G. Mlinžek, T. Šolmajer, J. Zupan, M. Novič, *J. of Chemometrics,* **2007**, *21* (7/9), 346–356.
5. J. Zupan, *Kemometrija,* KI in NR, Ljubljana, **2009**, pp. 267–272.
6. *Handbook of Genetic Algorithms,* Ed. L. Davis, Van Nostrand Reinhold, New York, **1991,** Chapter 6, pp. 72–90.
7. J. Zupan, *Kemometrija,* KI in NR, Ljubljana **2009**, pp. 217–249.
8. Š. Župerl, *Chemometric treatment of structure property relationship for the design and transfer of drugs into cells*, PhD. Thesis, University of Ljubljana, **2010**.
9. A. R. Katritzky, V. S. Lobanov, M. Karelson, CODESSA reference manual, 2.0, Gainsville. **1994**
10. M. Karelson, V. S. Lobanov, A. R. Katritzky, *Chem. Rev.,* **1996**, *96*, 1027–1043

Povzetek

Na nizu 53 tripsinsih inhibitorjev smo z linearnimi (MLR) in nelinearnimi metodami (ANN) modelirali njihovo afiniteto do kovalentno vezanih ligandov. Vsaka spojina je bila predstavljena s 343 molekulskimi deskriptorji. Preverjali smo hipotezo, da linearno modeliranje (MLR) zaradi premajhnega števila spojin v učnem nizu ne nudi možnosti izbire tolikšnega števila deskriptorjev, da bi to zadostovalo, za izdelavo dovolj dobrega modela. Po drugi strani pa modeliranje s protitočnimi nevronskimi mrežami (CP ANN) nima te omejitve in zaradi tega lahko pri njej uporabimo predstavitve spojin z večjim številom deskriptorjev, kot je število spojin v učenem nizu. Obe modelni metodi sta bili uporabljeni na povsem enaki delitvi niza 53 spojin na tri skupine, na učno, testno in validacijsko. S pomočjo genetskega algoritma (GA) smo preiskali vse možne oblike linearnih polinomov, ki jih dovoljuje velikost učnega niza, tj., vse velikosti sistema enačb s tremi do petindvajsetimi deskriptorji. Sistematičen pregled z modeli narejenimi z metodo MLR ni dal boljšega modela od tistega, ki jo je dal CP ANN model.