Journal of Computer-Aided Molecular Design (2005) 19: 137–147
DOI 10.1007/s10822-005-3321-5

An *in silico* approach for screening flavonoids as P-glycoprotein inhibitors based on a Bayesian-regularized neural network

Yong-Hua Wang[a], Yan Li[b], Sheng-Li Yang[a] & Ling Yang[a,*]
[a]*Laboratory of Pharmaceutical Resource Discovery, Dalian Institute of Chemical Physics, Graduate School of the Chinese Academy of Sciences, No. 457 Zhongshan Road, 116023, Dalian, China;* [b]*School of Chemical Engineering, Dalian University of Technology, No. 158 Zhongshan Road, 116012, Dalian, China*

Received 11 October 2004; accepted in revised form 3 March 2005

Key words: back-propagation neural network, Bayesian-regularized neural network, flavonoid, log K_d, partial least squares analysis, P-glycoprotein, quantitative structure–activity relationship

Summary

P-glycoprotein (P-gp), an ATP-binding cassette (ABC) transporter, functions as a biological barrier by extruding cytotoxic agents out of cells, resulting in an obstacle in chemotherapeutic treatment of cancer. In order to aid in the development of potential P-gp inhibitors, we constructed a quantitative structure–activity relationship (QSAR) model of flavonoids as P-gp inhibitors based on Bayesian-regularized neural network (BRNN). A dataset of 57 flavonoids collected from a literature binding to the C-terminal nucleotide-binding domain of mouse P-gp was compiled. The predictive ability of the model was assessed using a test set that was independent of the training set, which showed a standard error of prediction of 0.146 ± 0.006 (data scaled from 0 to 1). Meanwhile, two other mathematical tools, back-propagation neural network (BPNN) and partial least squares (PLS) were also attempted to build QSAR models. The BRNN provided slightly better results for the test set compared to BPNN, but the difference was not significant according to *F*-statistic at $p = 0.05$. The PLS failed to build a reliable model in the present study. Our study indicates that the BRNN-based *in silico* model has good potential in facilitating the prediction of P-gp flavonoid inhibitors and might be applied in further drug design.

Abbreviations: BPNN – back-propagation neural network; BRNN – Bayesian-regularized neural network; E-state – electrotopological state; NBD2 – C-terminal nucleotide-binding domain; PCA – principal component analysis; P-gp – P-glycoprotein; PLS – partial least squares; QSAR – quantitative structure–activity relationship

Introduction

P-glycoprotein (P-gp) is a 170-kDa transmembrane protein found in various resistant tumor cells [1, 2]. P-gp belongs to the ABC (ATP-binding cassette) transporter family as it actively transports a broad spectrum of structurally and functionally dissimilar drugs out of the cell and, in this way, limits the efficacy of many drugs. Therefore, the overexpression of P-gp and consequent multidrug resistance (MDR) phenotype are major obstacles to successful cancer chemotherapy [3, 4]. In the last years great efforts have been made to discover effective and nontoxic drugs that are able to reverse P-gp mediate-drug efflux [5].

Flavonoids are a group of polyphenolic compounds particularly abundant in fruits, vegetables, nuts, stems, flowers or plant-derived beverages like wine and tea, and constitute important components

*To whom correspondence should be addressed. Fax: +86-0411-84676961; E-mail: yling@dicp.ac.cn

of normal human food, with a few hundreds of milligrams daily intake in human diet [6]. According to Harborne and Williams [7] a total of more than 6500 different flavonoids have been identified from plant sources since 1992. Flavonoids display a remarkable variety of biochemical and pharmacological properties believed to be beneficial to human health, including antioxidant, antiviral, anticarcinogenic, and anti-inflammatory activities [8]. For example, some flavonoid analogs have been demonstrated to inhibit the growth of various cancer cell lines *in vitro* as well as to reduce the tumor development in experimental animals [9]. Up to now many investigations have been focused on identifying the interaction of flavonoids with P-gp, and have successfully proved the inhibitory effects of some, but not all, flavonoid derivatives on P-gp-mediated drug transport [8, 10–14]. Quercetin and a methoxylated derivative inhibited rhodamine 123 effluxes and reverted MDR in MCF-7 breast cancer cells [10, 11]. Biochanin A, morin, phloretin, and silymarin have also been confirmed to inhibit P-gp-mediated cellular efflux and the mechanism of the interaction involved, at least in part a direct interaction [12]. Isoflavone and flavone compounds were demonstrated to be active against P-gp [13]. Most of the 28 *n*-benzylpiperazine flavonoids synthesized by Jacques et al. [14] displayed MDR-modulating activity, some of them being more potent than the calcium-channel blocker verapamil, a standard MDR-reversing agent. Prenylated flavonoids have also been demonstrated to bind to P-gp with high affinity, and strongly inhibit drug interaction and nucleotide hydrolysis [8].

However, to our knowledge, an *in silico* model as potentially facile and economic alternative to *in vitro* methods to evaluate the inhibitory effects of flavonoid compounds on P-gp is still unavailable. The use of computational models over their traditional counterparts is preferred now for many reasons like ease of use, speed and relatively low cost. Up to now the atomic resolution structure of P-gp is still lacking, little is known about the three-dimensional structure of this transmembrane protein [15]. All these have made the efforts be primarily directed at the development of QSAR models [16, 17], as well as the identification of pharmacophoric features for prediction of P-gp substrates [17] and inhibitors [18]. The aim of this study, therefore, was to quantitatively identify the relationships between the chemical structures and anti-MDR (P-gp mediated) activities of flavonoids.

During the development of QSAR modeling, various fitting methods such as artificial neural networks (ANN)[19, 20], standard and partial least squares (SLS and PLS) [21, 22] have been used to analyze the structure–activity data to predict chemical properties, and so far they have exhibited wide applications in QSAR analysis [22, 23]. More recently, Bayesian concepts and methodology have gained popularity as a tool for facilitating drug discovery and development process [24], though many years before they have existed [25, 26]. The present study is undertaken to use the Bayesian-regularized neural network (BRNN) in establishing QSAR models for screening flavonoids as P-glycoprotein inhibitors. As a comparison to BRNN, two alternative algorithms, i.e., the back-propagation neural network (BPNN) and PLS were also attempted in our study since they are frequently used in QSAR analyses.

Materials and methods

Dataset description

A series of 57 structurally diverse flavonoid derivatives binding to the C-terminal nucleotide-binding domain (NBD2) of mouse P-gp were collected from a recent paper [8] (Table 1). The potency of flavonoids as P-gp inhibitors was estimated by the binding ability measured by the K_d values, for flavonoids with high binding affinity to the NBD2 of P-gp are able to increase the intracellular drug accumulation [8]. Their study was performed using a four-step procedure: (i) direct binding to purified recombinant cytosolic mouse NBD and/or full-length transporter, (ii) inhibition of ATP hydrolysis and energy-dependent drug interaction with transporter-enriched membranes, (iii) inhibition of cell transporter activity monitored by flow cytometry and (iv) chemosensitization of cell growth [8]. In the present work, we try to use this group of flavonoids as datasets to build potent QSAR models for prediction of their biological properties, i.e., the log K_d.

Molecular modeling

In this work, the molecular modeling was carried out using SYBYL 6.92 (Tripos Associates. St.

Table 1. Flavonoid inhibitors of P-gp with the binding affinity of log K_d values used in this study.

No.	Flavonoid	log K_d (μM)	No.	Flavonoid	log K_d (μM)
1	3-OH-flavone	1.004	30	8-DMA-galangin	−0.347
2	7-OH-flavone	1.543	31	8-DMA-kaempferide	−0.699
3	4-Chloro-chalcone	0.114	32	Chrysin	0.949
4	4-Bromo-chalcone	−0.244	33	6-Prenyl-galangin	−0.658
5	4-Fluoro-chalcone	0.556	34	Broussochalcone A	−0.357
6	4-n-$C_{10}H_{21}$-chalcone	−1.222	35	Chalcone	0.663
7	4-Iodo-chalcone	−0.602	36	Apigenin	1.004
8	4-Methoxy-chalcone	0.362	37	8-DMA,3,7-dimethyl-galangin	−0.824
9	4-*n*-C_2H_5-chalcone	0.415	38	2′,4′-Dichloro-galangin	0.602
10	4-*n*-$C_{14}H_{29}$-chalcone	1.152	39	Galangin	0.724
11	4-*n*-C_4H_9-chalcone	0.000	40	4′-Iodo-galangin	0.041
12	4-*n*-C_6H_{13}-chalcone	−0.567	41	4′-*n*-C_8H_{17}-galangin	−1.222
13	4-n-C_8H_{17}-chalcone	−1.699	42	4′-Fluoro-galangin	0.833
14	4-OH-chalcone	0.681	43	6-Geranyl-dehydrosilybin	−0.745
15	4-*n*-Cyclohexyl-chalcone	−0.276	44	8-Geranyl-dehydrosilybin	−0.921
16	4-OH-3-prenyl-chalcone	−0.276	45	Dehydrosilybin	0.342
17	6-Methyl-chrysin	0.491	46	6-Prenyl-dehydrosilybin	−0.432
18	6,7-Dimethyl-chrysin	0.114	47	8-Prenyl-dehydrosilybin	−0.602
19	6-Geranyl-chrysin	−1.347	48	Silybin	0.833
20	6-Benzyl-chrysin	−0.469	49	Kaempferide	0.653
21	6-Prenyl-chrysin	−0.523	50	Kaempferol	0.826
22	3-Methyl-galangin	0.949	51	Naringenin	1.562
23	6-Prenyl-galangin	−0.678	52	Quercetin	0.845
24	7-*O*-isopropyl-chrysin	0.114	53	Tectochrysin	0.799
25	8-DMA-apigenin	−0.155	54	3′,4′-Difluoro-chrysin	0.799
26	8-Benzyl-chrysin	−0.004	55	4′-Iodo-chrysin	0.342
27	8-Prenyl-chrysin	−0.553	56	Taxifolin	1.573
28	8-DMA-chrysin	−0.699	57	Genistein	1.423
29	8-Geranyl-chrysin	−1.602			

Louis, MO) and Dell precision 650 running the LINUX RedHat 8.0 operating system. All two-dimensional structures of the compounds were obtained from a commercial available MDL-ISIS database and the literature [8].

Calculation of molecular descriptors

Construction of the QSAR models firstly depends on the generation of molecular descriptors. By simply using various molecular modeling tools, it is possible to calculate thousands of these descriptors directly from the structure of any particular molecule [27]. Molconn-Z program (in the SYBYL software package) is a useful tool to calculate the molecular connectivity indices and electrotopological state (E-state) descriptors. Molconn-Z extends the descriptor set produced by its predecessor Molconn-X, which calculates E-state hydrogen and bond-type descriptors and provides 462 descriptors. Molconn-Z calculates additional 287 descriptors. These variables of molecular structure include: the molecular connectivity Chi indices, $^{m}\chi_t$ and $^{m}\chi_t^v$; Kappa shape indices, ^{m}k and $^{m}k_\alpha$; E-state indices, ES_i; hydrogen E-state indices, HES_i; atom type and bond type E-state indices; topological equivalence indices and total topological index.

Although this program calculates more than 700 molecular descriptors, for the present example, we selected the 'standard Molconn-Z descriptors' that satisfied two requirements: (i) they were physically intuitive and directly related to gross structural and/or molecular bio-physical properties, and (ii) they were a relatively small number of molecular descriptors but most useful in QSAR

modeling (Molconn-Z manual), including 248 descriptors. The rationale for this decision was that the aim of the present study was to develop a reliable QSAR model but using as few as possible descriptors.

CLogP values, i.e., the log of the 1-octanol/water partition coefficient parameters frequently used in QSAR studies, were also applied in this work. CLogP values for this dataset of flavonoids were obtained from MDL-ISIS database or calculated from web [28].

Thus so far a total of 249 descriptors for each molecule, including 248 Molconn-Z descriptors and one CLogP value were calculated in current study.

Variable reduction

Before generating QSAR models, it is often necessary to perform a variable reduction process on the original set of descriptors [29, 30]. Basically, the objective of variable selection is threefold: (a) to provide faster and more cost-efficient predictors; (b) to provide a better understanding of the underlying process that generated the data; (c) also the most important one is, to eliminate noise (uninformative descriptors) and prevent overfitting or chance correlations. For these reasons, several procedures shown as follows were applied to reduce the number of descriptors.

(i) Descriptors with constant values as well as descriptors containing 95% zero values were removed. After these steps, the original 249 descriptors were reduced to 67 ones (Table 2). These descriptors were further reduced to 53 by removing those ones (shown with superscript in Table 2) with the standard deviation less than 0.5. These descriptors were removed because they offered little information for the construction of the models.

(ii) To further reduce the chance of correlation among descriptors and the descriptor space, a principal component analysis (PCA) was performed. Eventually, four principal component vectors were constructed and used further as input parameters in neural network experiments. Before the PCA process, the given dataset (including structural feature data and activity data) was normalized so that the input and output variables would have means of zero and standard deviations of one (Table 2).

Regression methods

The present QSAR models attempted to correlate the target activity (log K_d) with molecular descriptors using Bayesian-regularized neural network. For comparison, we also attempted two other methods, i.e., classical BPNN and partial least squares analysis to build QSAR models. Here we provide only a summary of these methods as details appear elsewhere (Refs. [31–33] for BRNN theory, Ref. [34] for BPNN and Ref. [22] for PLS).

The Bayesian framework for neural networks is based on a probabilistic interpretation of network training to improve generalization ability of the classical neural networks [31–33]. In contrast to conventional network training, where an optimal set of weights is chosen by minimizing an error function, the Bayesian approach involves a probability distribution of network weights [31–33]. As a result, the predictions of the network are also probability distributions. Most importantly, complex models are penalized in the Bayesian approach, reducing the problems of overfitting and overtraining [31–33]. In this work, the three-layer networks were fully connected, with a hyperbolic tangent function employed in the hidden layer and a linear transfer function for the output layer. The network used the Bayesian regularization [31, 32] to find out the optimum weights for the network, and the Levenberg–Marquardt training algorithm [34] was applied to accelerate convergence of targets. The starting-values for the parameters of the BRNN model were selected according to the Nguyen–Widrow rule [35]. The training is stopped at the maximum of the evidence maximum for the hyperparameters α and β (α representes the weight decay regularization, while β governs the variance of the noise) [31–33]. The obtained BRNN model for predicting log K_d had a 4-6-1 architecture. In this work, we used an internally developed C language program mainly based on paper by Foresee and Hagan [36].

Classical BPNN is the most prevalent of the supervised learning models of neural network [34], frequently applied in QSAR studies [37]. In present study, a three-layer fully connected

Table 2. Molecular indices used in the QSAR studies.

Index	Definition	Index	Definition
CLogP	Log of 1-octanol/water partition coefficient	nsOH	Number of –OH
Xv0-Xv2	Connectivity simple path indices	nHCsatu	Number of chn (unsaturated)
Xvp3-Xvp5	Connectivity simple path indices	nsCH3	Number of -CH_3
Xvp6*-Xvp10*	Connectivity simple path indices	nssCH2	Number of –CH_2–
Xvc3*	Connectivity simple cluster indices	ndsCH	Number of =CH–
Xvpc4	Connectivity simple path/cluster-4 index	naaCH	Number of :CH:
Xvch6	Connectivity valence chain indices	ndssC	Number of =C<
ka_1, ka_2, ka_3	Kappa alpha indices	naasC	Number of :C:-
phia	Flexibility index (k_1*k_2/nvx)	SsF	E-state index values for atom type
nvx	Number of nonH atoms in molecule	SdO*	E-state index values for Keto oxygens
nedges	Number of edges (bonds) in the molecule	SssO*	E-state index values for ether oxygen
nrings	Number of rings	SHsOH	E-state index values for –OH
sumdelI	Sum of delta *I* values (used in calculation of electrotopological state (E-state) index)	SHaaCH	E-state index values for :CH:
sumI	Sum of intrinsic state values *I* (used in calculation of E-state index)	SHCsats	E-state index values for CH_n (saturated)
Qv*	General polarity descriptor	SHCsatu	E-state index values for CH_n (unsaturated)
nHBd	Hydrogen bond (H-bond) donor counts	SsCH3	E-state index values for –CH_3
nHBa	H-bond acceptor counts	SssCH2	E-state index values for –CH_2
nwHBa	Weak hydrogen acceptor counts	SHdsCH	E-state index values for =CH–
SHBd	Donor descriptor for molecule (sum of hydrogen E-state values for all H-bond donors in the molecule). The following groups are classified as donors: –OH, =NH, –NH_2, –NH–, –SH, and CH	SHBa	H-bond acceptor descriptor for molecule (sum of estate values for all H-bond acceptors in the molecule). The following groups are classified as acceptors: –OH, =NH, –NH_2, –NH–, >N–, –O–, =O, and –Cl
SwHBa	Descriptor for weak H-bond acceptor (sum of E-state values for all weak hydrogen bond acceptors)	SdssC	E-state index values for *s* =C<
H_{max}	Largest hydrogen E-state value	SaasC	E-state index values for :C:-
G_{max}*	Largest E-state value	SssssC	E-state index values for >C<
H_{min}*	Smallest hydrogen E-state value	SsOH	E-state index values for –OH
G_{min}*	Smallest E-state value	SdsCH	E-state index values for =CH–
nHsOH	Number of –OH	SaaCH	E-state index values for:CH:
nHdsCH	Number of =CH–	SsssCH	E-state index values for methylenes
NssO	Number of –O–	SHdCH2*	E-state index values for =CH_2
nHaaCH	Number of :CH:	Sketone*	E-state index values RC(=O)R
nHCsats	Number of CHn (saturated)		

*Descriptors with standard deviation <0.5.

network based on the Levenberg–Marquardt training algorithm [34] was applied. A Tan-sigmoid function and a linear transfer function were used in the hidden and output neurons, respectively. When building the BPNN model, an additional validation set containing 25% of the whole dataset was applied to control the training process. The training would not stop until the training errors approached to zero and the validation set showed no further performance improvement. The final optimal BPNN for modeling log K_d had a 4-6-1 architecture, which was then tested using another subset that had not been used in training the network. An internally developed C language program of BPNN was used for the QSAR analysis.

PLS is a popular computational method that expresses a dependent variable (target activity) in terms of linear combinations of the independent variables commonly known as principal components (PCs). It is similar to principal component regression but with both the independent and dependent variables involved in the generation of the orthogonal latent variables rather than only independent variables [35]. PLS is an iterative algorithm with consecutive estimates obtained using the residuals from previous iterations as the new dependent variable [38]. Each iteration of the algorithm introduces another latent variable, and leave-one-out cross-validation was used to determine the number of components that yields an optimally predictive model. The number of latent variables was chosen to maximize cross-validated R^2 (called Q^2) of the training set. The model is generally considered internally predictive if $Q^2 > 0.5$ [39], as generally the Q^2 are much better indicators than standard error and conventional R^2 of how reliable predictions actually are [39]. Following this internal validation, the model was evaluated externally using a test set of flavonoids that were not used in training the QSAR model. In present study, the PLS analysis was performed by using QSAR module in SYBYL software package.

Building procedure of the models

For QSAR modeling, the following steps were taken:

(i) The initial chemical structure of each molecule was constructed by molecular modeling using SYBYL software.
(ii) The dataset was consolidated into a single file (the SYBYL structure–data format, *tbl*) which contained structural and activity data (log K_d values) for each molecule.
(iii) By using a *K*-means clustering algorithm on the X (indices) and Y (log K_d) values taken together, the data set was divided into a training, a validation and a test set. The test set of 25% of the total number of compounds was built by randomly choosing one-fourth from each cluster. The test data were not used in training the regression models. Because the training set should cover the whole range of log K_d values to avoid extrapolation during the final prediction, we put the maximum and the minimum log K_d values in this dataset.
(iv) BRNN, BPNN and PLS algorithms were employed to build the respective QSAR models. In order to make the comparisons of performances of the three methods on common ground, the PCA-NN and PLS models employed the same test data with the same 53 input molecular indices. With an optimal number of PCs and architecture, the NN models were trained independently 25 times to eliminate spurious effects caused by the random set of initial weights.

Results and discussion

Molecular descriptors

The principal component analysis has been performed on 53 components (descriptors). As a result, four principal components (PCs) shown in Table 3 were obtained, the number of which was determined by the standard error of prediction of the test set rather than by the minimum variance described by the PCs. It was found that the four PCs suffice to explain 85% of variance and represent the predominant information of the original descriptor set, which were sufficient in all network-modeling cases reported here. Therefore, in this way, the original 53 descriptors were eventually reduced to four PCs, thus eliminating a great number of uninformative information as

Table 3. Descriptive statistics for the four principal components.

	PC1	PC2	PC3	PC4
Eigenvalue	19.99	6.50	4.89	3.26
% Variance explained	0.48	0.16	0.12	0.08
Total % variance explained	0.48	0.64	0.76	0.85

well as the chance of correlation, which will be helpful for increasing the speed and accuracy of the QSAR modeling.

BRNN, BPNN and PLS

There are many regression methods available to find the best correlation between descriptors and the variables to be explained. In this work, three methods were tried in order to build reasonable predictive models. Two methods are based on neural networks, the Bayesian neural nets and the classical back-propagation neural nets; one other method is partial least squares methodology based on linear regressions. The statistical results of the optimum NN and PLS models developed in this work are summarized in Table 4.

The optimum BRNN model for the present data has a 4-6-1 architecture. For the training set the standard error of estimation (SEE) is 0.120 ± 0.006 (data scaled from 0 to 1) and the standard error of prediction (SEP) for the testing set is 0.146 ± 0.006. The conventional coefficients R^2 for training set and testing set are 0.756 ± 0.061 and 0.728 ± 0.072, respectively. The SEP of the test data is of the same order of the magnitude as the SEE of the training data, indicating the model is not overfitted. Figure 1a and b shows the performance of the BRNN model for the training data and test data, respectively. From these results, it can be concluded that the present method, i.e., the BRNN, seems to be predictive from both an internal (training) and an external point of view with respect to the predictions of the test sets.

The BPNN with best performance has a 4-6-1 architecture, presenting a SEE = 0.119 ± 0.041 for the training set, a SEE = 0.175 ± 0.036 for validation set and a SEP = 0.160 ± 0.042 for test set. Since these results are not substantially worse than the results obtained by BRNN model, the BPNN model is still reasonable. In conclusion, the above results have demonstrated the capability of both two neural networks, the BRNN and BPNN, in modeling the log K_d values.

Whereas, could PLS, the widely used linear regression method in QSAR modeling, still be applied in building a reliable model for the present study? With this question, the PLS regression was carried out for the dataset with the 53 variables, resulting in a 1-latent variable QSAR model with the conventional correlation coefficients $R^2 = 0.392$ and cross-validated $Q^2 = 0.144$ for the training set. The SEE for the training data is 0.657 and the p-value is 0.000 using F-test. All the data show that the model is internally bad predictive. The model was also evaluated on fresh data, i.e., the test data, and the R^2 is 0.655 and the SEP is 0.267. In a word, PLS generated a relatively poor QSAR model for P-gp flavonoid inhibitors.

Comparisons of the three regression algorithms

In order to compare the performances of the different models the F-test [40] was applied in this work.

Table 4. Statistical results of the three optimal QSAR models for P-gp flavonoid inhibitors by each regression method.

Parameter	BRNN		BPNN			PLS	
	Training	Test	Training	Validation	Test	Training[a]	Test
Size	43	14	43	14	14	43	14
Num/Pc[b]	4	4	4	4	4	1	1
R^{2c}	0.756 ± 0.061	0.728 ± 0.072	0.826 ± 0.130	0.721 ± 0.132	0.679 ± 0.079	0.392	0.655
SEE[d]	0.120 ± 0.006	–	0.119 ± 0.041	–	–	0.657	–
SEP[e]	–	0.146 ± 0.006	–	0.175 ± 0.036	0.160 ± 0.042	–	0.267

–, not applicable or available.
[a]Training using leave-one-out cross-validation procedure.
[b]Num/Pc, number of principle components.
[c]R, the conventional coefficient.
[d]SEE, standard error of estimate (data scaled).
[e]SEP, standard error of prediction (scaled).

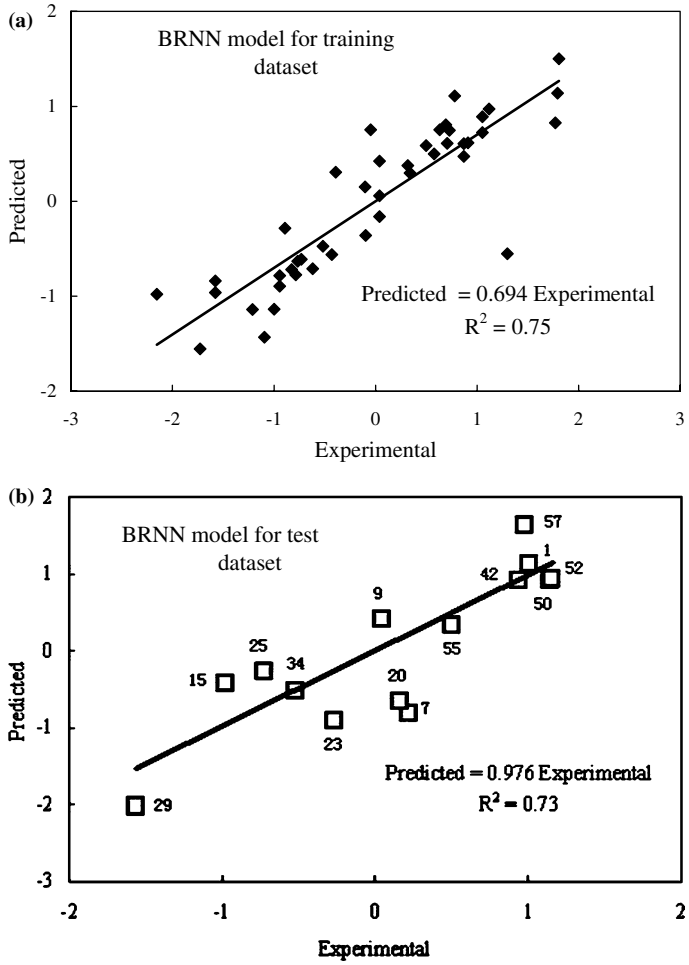

Figure 1. Predicted vs. experimental log K_d values (μM) of 57 flavonoids by Bayesian-regularized neural network for (a) the training set and (b) the test set. The original log K_d data have been scaled from 0 to 1.

$$F(n_1, n_2) = \mathrm{SEP_1}^2/\mathrm{SEP_2}^2$$

Where, n_1 and n_2 are the number of samples in the test set, $\mathrm{SEP_1}^2$ is the square from the higher and $\mathrm{SEP_2}^2$ is the square from lower root mean square errors of the two compared models. When comparing the performances of PLS with both NNs, *F*-values are 3.34 and 2.78 for PLS/BRNN and PLS/BPNN, respectively, which are higher than the critical one (2.48), indicating statistically significant difference at a level of significance of 0.95. As for the two NN models, the calculated *F*-value (1.20) is lower than the critical one (2.48). The results show that at level of significance of 0.95 the differences in the performances of the mean SEP in both network architectures differ only randomly. Table 5 summarizes the statistical results of the three models. In summary, the statistical tests reveal that the BPNN is comparable to the BRNN model, but the performances of NN and PLS methods are quite different.

PLS is a useful regression tool commonly used in QSAR analysis. It is a linear technique, and thus determination of the relative importance of descriptors is possible. In this work, a reliable quantitative structure-log K_d relationship cannot be derived by the PLS analysis. The failure might be due to that linear regression method often makes models unstable and inaccurate when dealing with data of complex nonlinear relationships. As compared with traditional statistic methods, e.g., the PLS, methods based on neural networks are often slow to train and do not allow easy model interpretation of

Table 5. Comparison of different models using *F*-test (at level of significance 0.95).

Compared models	Calculated *F*-value	Calculated *F*-value
PLS/BRNN	3.34	2.48
PLS/BPNN	2.78	2.48
BRNN/BPNN	1.20	2.48

what should be changed in a structure to improve the predicted property. However, neural nets have the advantage of being able to model nonlinear relationships between the dependent and the independent variables [41], even without prior knowledge of the form of nonlinearity. However, overtraining and overfitting problems often limit their applications in generating QSAR models. A modified form of NN, the Bayesian regularized artificial neural network, is more resistant to overfitting and overtraining than classical NN, resulting in improved performance of generalization [42].

BPNN has demonstrated its ability to model log K_d values for the present dataset. However, BPNN still presents some problems, such as optimization of neural network architecture, selection of best models, overfitting and overtraining [41]. Overfitting occurs when too many weights are used compared to the number of compounds available in the training set. For a three layered fully connected network, one method to reduce overfitting is to keep the number of weights under half the number of examples of the training set (a rule of thumb). Another generally used method to reduce overfitting and overtraining is the introduction of a validation set during training network, which was also used in our networks.

In this work, the comparisons between the BP and BR neural networks have lead to some interesting findings (Table 4). Each NN model with the 4-6-1 architecture has been trained 25 times independently. For BPNN model, there are 15.7% variation of R^2 for the training and 18.3% variation for the validation set, as well as 12% variation for the test set. Whereas, the BRNN model presents only about 8% of variation for the training and 9% of variation for test set. Moreover, the standard deviation of SEE and SEP for both the NN models are significantly different. For BRNN the deviation is 0.006, which is only 15% of the average deviation (0.04) of the three sets (the training, validation and test sets) calculated by BPNN. Compared with BR neural nets, BP network reveals relatively big deviation of regression when modeling the training and test data. The main reasons for the phenomenon of BPNN mentioned above may be summarized as follows. On one hand, the prediction results of BPNN depend on the initial random distribution of the parameters (weights) which define the neural nets, thus leading to variation of modeling results by a network every time. On the other hand, BPNN is lack of self-organizing capacity to automatically select new data with large information and to update its weights to well match the new data [31]. These factors offered further support the superiority of BRNN to BPNN.

Bayesian probability methods are complementary to the conventional networks as their shortcomings can be overcome to automatically control model complexity [31, 32, 43]. BRNN differs from conventional networks in that the every weight is replaced by a distribution of weights. However, the classical BPNN uses a single set of weights to make predictions of new data, limiting its ability of generalization. In conventional neural networks, different models with different connected weights may share the similar SEPs. However BRNN model may give the similar SEPs with similar weights [41] (Table 4). BRNN applies a posterior distribution over network weights that will give rise to a distribution over the outputs of the network for some new case, so that a local minimum is likely to be avoided. Thus BRNN allows evaluation of the likely uncertainty of a prediction [44]. However, both PLS and BPNN algorithms often converge to a local minimum, preventing them from generalizing ability. Recently, Bayesian regularization has been demonstrated as an effective method to solve the overfitting and overtraining problems of conventional networks [43], the BRNN algorithm is robust.

In this work, in order to evaluate the robustness of the BRNN model, different numbers of hidden neurons from 5 to 10 for the net was adopted, and all the resulted standard deviations of SEP and SEE were less than 0.1 (data not shown). However, for BP networks different numbers of hidden neurons can significantly influence the performance of the network [41], optimization of neural network architecture is often key to classical network modeling. In addition, another analysis was also performed to assess the quality of the model by evaluating the standard deviations of SEE of the training data and SEP of the test data. When the training was repeated 25 times on the 4-6-1 model, the net gives very slight fluctuations of prediction results for both the training and test sets (less than 0.01) (Table 4). The results show the strong robustness of BRNN, which is consistent with a previous study [41].

Another marked advantage of Bayesian neural network is that no validation or test is involved, but the net is still robust with optimal predictions [31, 32, 45]. Recently, Roberts and coworkers [46] have shown that the performance of Bayesian network is almost the same for training and for testing set. In our study, the BRNN model exhibited significant ability of generalization for new data (the test set) though no validation set was involved. Several studies have demonstrated that in BRNN modeling the SEE is sufficient to estimate the network's generalization performance [41, 46], while for a small dataset an additional test set may be helpful to validate the network behavior. Therefore, 25% of the data that were never used in training were adopted as the test set to evaluate the model. The variation is within 3% of magnitude between the SEE and the SEP of the proposed BRNN model (Table 4). The results clearly exhibit an excellent predictive power of the system for external dataset.

In this study, the results of BRNN model are indicative of its abilities to accommodate nonlinearities in the activity and structural descriptors. The advantages of BRNN, such as robustness, no additional test requirement, and optimal predictivity were validated in our work.

Conclusion

To establish an *in silico* model for screening flavonoid P-gp inhibitors using fundamental molecular descriptors, the BRNN approach was attempted in the present work. It can be concluded from our results that only by using simple calculated chemical descriptors a useful BRNN-derived QSAR model can be built with satisfactory performance. Since the indices used in this model are easily calculated and do not rely on any measured values, the model is of proper practicality and will facilitate further development of potent flavonoid P-gp inhibitors.

Meanwhile, the comparison of this BRNN-derived model and the other two models established by BPNN and PLS, also two widely used methods in QSAR modeling, was conducted. The comparison results have demonstrated the superiority of BRNN to both BPNN and PLS in this study, further proving the feasibility, reliability and robustness of BRNN in modeling the log K_d values for P-gp flavonoid inhibitors.

Acknowledgements

We thank the 973 Program (2003CCA03400) and the 863 Program (2003AA223061) from the Ministry of Science and Technology of China, and DICP Innovation Fund of the Chinese Academy of Sciences.

References

1. Cordon-Cardo, C., O'Brien, J.P., Boccia, J., Casals, D., Bertino, J.R. and Melamed, M.R., J. Histochem. Cytochem., 38 (1990) 1277.
2. Thiebaut, F., Tsuruo, T., Hamada, H., Gottesman, M.M., Pastan, I. and Willingham, M.C., Proc. Natl. Acad. Sci., 84 (1987) 7735.
3. Thiebaut, F., Tsuruo, T., Hamada, H., Gottesman, M.M., Pastan, I. and Willingham, M.C., J. Histochem. Cytochem., 37 (1989) 159.
4. Yu, D.K, J Clin Pharmacol., 39 (1999) 1203.
5. Wiese, M. and Pajeva, I.K., Curr. Med. Chem., 8 (2001) 685.
6. Hollman, P.C.H. and Katan, M.B., Food Chem. Toxicol., 37 (1999) 937.
7. Harborne, J.B. and Williams, C.A., Phytochemistry, 55 (2000) 481.
8. Pietro, A.D., Conseil, G., Perez-Victoria, J.M., Dayan, G., Baubichon-Cortay, H., Trompier, D., Stein-fels, E., Jault, J.M., de Wet, H., Maitrejean, M., Comte, G., Boumendjel, A., Mariotte, A.M., Dumontet, C., McIntosh, D.B., Goffeau, A., Castanys, S., Gamarro, F. and Barron, D., Cell. Mol. Life Sci., 59 (2002) 307.
9. Versantvoort, C.H.M., Schuurhuis, G.J., Pinedo, H.M., Eekman, C.A., Kuiper, C.M., Lankelma, J. and Broxterman, H.J., Br. J. Cancer, 68 (1993) 939.

10. Scambia, G., Ranelletti, F.O., Panici, P.B., De Vincenzo, R., Bonanno, G., Ferrandina, G., Piantelli, M., Bussa, S., Rumi, C. and Cianfriglia, M., Cancer Chemother. Pharmacol., 34 (1994) 459.
11. Shapiro, A.B. and Ling, V., Biochem. Pharmacol., 53 (1997) 587.
12. Zhang, S. and Morris, M.E., J. Pharmacol. Exp. Ther., 304 (2003) 1258.
13. Chieli, E., Romiti, N., Cervelli, F. and Tongiani, R., Life Sci., 57 (1995) 1741.
14. Ferté, J., Kühnel, J.M., Chapuis, G., Rolland, Y., Lewin, G. and Schwaller, M.A., J. Med. Chem., 42 (1999) 478.
15. Safa, A.R., Curr. Med. Chem. Anti-Canc. Agents., 4 (2004) 1.
16. Bain, L.J., McLachlan, J.B. and LeBlanc, G.A., Environ. Health Perspect., 105 (1997) 812.
17. Li, Y., Wang, Y., Yang, L., Zhang, S., Liu, C. and Yang, S., J. Mol. Struct., 733 (2005) 111.
18. Ekins, S., Kim, R.B., Leake, B.F., Dantzig, A.H., Schuetz, E.G., Lan, L.B., Yasuda, K., Shepard, R.L., Winter, M.A., Schuetz, J.D., Wikel, J.H. and Wrighton, S.A., Mol. Pharmacol., 61 (2002) 964.
19. Gasteiger, J. and Zupan, J., Angew.Chem. Int. Ed. Engl., 32 (1993) 503.
20. Burden, F.R., Quant. Struct.-Act. Relat., 15 (1996) 7.
21. Martin, Y.C. Quantitative Drug Design. Marcel Dekker, New York, 1978.
22. Ramsden, C.A., In Hansch, C. (Ed.), Comprehensive Medicinal Chemistry. Pergamon Press, New York, Vol. 4, 1990.
23. Dunn, W.J., In Clark, C.R. and Moos,W.H. (Ed.), Drug Discovery Technologies, Ellis Horwood Limited, New York, Chapter 2, 1990.
24. Winkler, D.A. and Burden, F.R., J. Mol. Graph. Model., 22 (2004) 499.
25. Bruneau, P., J. Chem. Inf. Comput. Sci., 41 (2001) 1605.
26. Polley, M. J., Winkler, D.A. and Burden, F.R., J. Med. Chem., 47 (2004) 6230.
27. Todeschini, R., Consonni, V., Mauri, A. and Pavan, M. Dragon Web version 3.0, Milano, Italy; 2003.
28. http://www.daylight.com/daycgi/ClogP.
29. Yu, H., Yang, J., Wang, W. and Han, J. Proc. IEEE Comput. Soc. Bioinformatics Conf. (CSB) (2003) 220.
30. Degroeve, S., De Baets, B., Van De Peer, Y. and Rouze, P., Bioinformatics, 18 (2002) S75.
31. MacKay, D.J.C., Neural Comput., 4 (1992) 415.
32. MacKay, D.J.C., Neural Comput., 4 (1992) 448.
33. MacKay, D.J.C., Comput. Neural Syst., 6 (1995) 469.
34. Hagan, M.T. and Menhaj, M.B., IEEE Trans. Neural Networks, 5 (1994) 989.
35. Nguyen, D. and Widrow, B., Proceedings of the International Joint Conference on Neural Networks, 3 (1990) 21.
36. Foresee , F.D. and Hagan, M.T., IEEE, International Conference on Neural Networks, 1997, 1930.
37. Terfloth, L. and Gasteiger, J., Drug Discovery Today, 6 (2001) 102.
38. Höskuldsson, A., J. Chemom., 2 (1988) 211.
39. Wold, S., Ruhe, A., Wold, H. and Dunn, W.J., J. Scientific Stat. Comput., 5 (1984) 735.
40. Bhandare, P., Mendelson Y., Peura, R.A., Janatsch, G., Kruse-Jarres, J.D., Marbach, R. and Heise, H.M., Appl. Spectrosc., 47 (1993) 1214.
41. Burden, F.R. and Winkler, D.A., J. Med. Chem., 42 (1999) 3183.
42. Bishop, C.M. Neural Networks for Pattern Recognition. Oxford University Press, New York, 1995.
43. Burden, F.R., Ford, M.G., Whitley, D.C. and Winkler, D.A., J. Chem. Inf. Comput. Sci., 40 (2000) 1423.
44. Neal, R.M. Bayesian Learning for Neural Networks. Springer, New York, 1996.
45. Buntine, W.L. and Weigend, A.S., Complex Syst., 5 (1991) 603.
46. Husmeier, Neural Networks., 12 (1999) 677.