Supervised Feature Ranking Using a Genetic Algorithm Optimized Artificial Neural Network

Thy-Hou Lin,* Shih-Hau Chiu, and Keng-Chang Tsai

Institute of Molecular Medicine & Department of Life Science, National Tsing Hua University,
Hsinchu, Taiwan 30013, R.O.C.

Received February 1, 2006

A genetic algorithm optimized artificial neural network GNW has been designed to rank features for two diversified multivariate data sets. The dimensions of these data sets are 85 × 24 and 62 × 25 for 24 or 25 molecular descriptors being computed for 85 matrix metalloproteinase-1 inhibitors or 62 hepatitis C virus NS3 protease inhibitors, respectively. Each molecular descriptor computed is treated as a feature and input into an input layer node of the artificial neural network. To optimize the artificial neural network by the genetic algorithm, each interconnected weight between input and hidden or between hidden and output layer nodes is binary encoded as a 16 bits string in a chromosome, and the chromosome is evolved by crossover and mutation operations. Each input layer node and its associated weights of the trained GNW are systematically omitted once (the self-depleted weights), and the corresponding weight adjustments due to the omission are computed to keep the overall network behavior unchanged. The primary feature ranking index defined as the sum of self-depleted weights and the corresponding weight adjustments computed is found capable of separating good from bad features for some artificial data sets of known feature rankings tested. The final feature indexes used to rank the data sets are computed as a sum of the weighted frequency of each feature being ranked in a particular rank for each data set being partitioned into numerous clusters. The two data sets are also clustered by a standard *K*-means method and trained by a support vector machine (SVM) for feature ranking using the computed *F*-scores as feature ranking index. It is found that GNW outperforms the SVM method on three artificial as well as the matrix metalloproteinase-1 inhibitor data sets studied. A clear-cut separation of good from bad features is offered by the GNW but not by the SVM method for a feature pool of known feature ranking.

INTRODUCTION

An important problem related to mining large data sets, both in dimension and size, is of selecting a subset of the original features. Preprocessing the data to obtain a smaller set of representative features, retaining the optimal salient characteristics of the data, not only decreases the processing time but also leads to more compactness of the models learned. When class labels of the data are available, we use supervised feature selection, otherwise unsupervised feature selection is appropriate.

Conventional methods of feature selection involve evaluating different feature subsets using some index and selecting the best among them. The index usually measures the capability of the respective subsets in classification or clustering depending on whether the selection process is supervised or unsupervised. A problem of these methods, when applied to large data sets, is the high-computational complexity involved in searching. The complexity is exponential in terms of the data dimension for an exhaustive search. Several heuristic techniques have been developed to circumvent this problem. Among them the branch and bound algorithm, suggested by Devijver and Kittler,[1] obtains the optimal subset in expectedly less than exponential computations when the feature evaluation criterion used is monotonic in nature. Greedy algorithms such as sequential forward and backward search[2] are also popular. These algorithms have quadratic complexity, but they perform poorly for nonmonotonic indices. In such cases, sequential floating searches[3] provide better results, though at the cost of a higher computational complexity. Beam search variants of the sequential algorithms[4] are also used to reduce computational complexity. Recently, robust methods for finding out the optimal subset for arbitrary evaluation indices are being developed using genetic algorithms (GAs).[5] GA based frature selection methods[6] are usually found to perform better than other heuristic search methods for large sized data sets; however, they also require considerable computation time for large data sets. Other attempts to decrease the computational time of feature selection include probabilistic search methods such as random hill climbing,[7] SCHEMATA+,[8] and Las Vegas Filter approach.[9]

The artificial neural networks (ANNs) have been grown in popularity and used extensively to perform various classification tasks. An ANN classifier performs a mapping from an input (feature or attribute) space onto an output (class) space. Cases are represented in the input space of ANN by a vector that contains the *n* feature values of a case. The output vector of the ANN is used to classify a case, e.g. by means of the winner takes all rule. An ANN with the optimal number of hidden nodes approaches a Bayesian classifier, and hence its error rate will be close to the

* Corresponding author fax: 886-03-575-3087; e-mail: thlin@life.nthu.edu.tw.

10.1021/ci0600354 CCC: $33.50 © 2006 American Chemical Society
Published on Web 06/17/2006

minimum error rate.[10] The ANNs can be divided into feedforward and recurrent classes depending on their connectivity. For many classification tasks, a large number of potentially useful features can be defined and added as input to an ANN. In these situations, feature selection is often a desired task. Ideally, when the acquisition costs of the features are equal, one wants to rank the available features according to the change in correctness that results from removing or adding the respective feature from the feature set.[11] The problem of feature selection using ANN can be seen as a special case of network pruning. The pruning of input nodes is equivalent to removing the corresponding features from the original feature set. Several pruning procedures for ANNs have been proposed,[12] but most of them focus on removing hidden nodes or connections, and they are not directly applicable to prune irrelevant input nodes. Pruning procedures extended to the removal of input nodes have been proposed by several groups,[13−18] where the variable selection process is typically based on a measure of the relevance of an input node, so that the less relevant features are removed. However, these techniques strictly rely on the adopted learning algorithm because the relevance of input nodes is evaluated during the training process.

In this report, we design an ANN to perform some feature ranking tasks based on the input node pruning process described previously by Castellano and Fanelli.[19] The feature selection tasks are conducted for 24 and 25 molecular descriptors computed respectively for 85 matrix metalloproteinase-1 (MMP-1) inhibitors[20] and 62 hepatitis C virus (HCV) NS3 protease inhibitors.[21,22] To ensure optimal partition on each data set, we use a K-means$^+$ algorithm[23] to split clusters by removing outliers from existing clusters to form new clusters. This will generate clusters where the initial assigned clustering states are not critical to the final clustering results. The clusters-labeled data sets are fed to the ANN where the training is optimized by a binary encoded GA, hence the method is designated as the GNW (genetic algorithm optimized neural network) method. The input nodes along with their connections of a trained GNW are successively removed (the self-depleted weights), and the remaining weights are adjusted in such a way that the overall input−output behavior learned by the network is kept approximately unchanged.[19] The sum of self-depleted weights plus the weight adjustments computed for each successively removed input node (feature) is used as a criterion to rank the features. It is found that the accuracy of the proposed feature ranking scheme is comparable to that trained by a support vector machine (SVM) method[24−27] on the same data sets tested. Finally, the actual feature rank of each data set is determined from counting frequency of each feature being ranked in a particular rank from a particular clustering of the data set. The feature rank thus obtained is independent of the effect how the data set is clustered.

MATERIALS AND METHODS

Procedures for constructing the structures of 85 MMP-1 inhibitors and 62 HCV NS3 protease inhibitors were similar to those described previously.[28] All 24 or 25 molecular descriptors were computed using the SYBYL[29] and Catalyst[30] suite of programs. The descriptor values of each data set were normalized to that between 1 and 0 to produce two data sets of dimension 85 × 24 and 62 × 25, respectively. The data sets were partitioned using a K-means$^+$ clustering scheme,[23] where the originally assigned K was autonomously adjusted by exploiting the statistical nature of the data for removing the degeneracy or empty clusters. The algorithm[23] uses the Euclidean distance to calculate the similarity between two objects. Each data set was partitioned into K clusters in the same way as K-means does, and then the K-means$^+$ algorithm splits clusters by removing outliers from existing clusters to form new clusters. The splitting threshold was 1.4σ, where σ was the standard deviation of the cluster. An object was considered as an outlier if its distance to the centroid of the cluster was greater than the splitting threshold. Once an outlier was found, it was removed from the current cluster and was assigned as the centroid of a new cluster. Some clusters were merged in the linking process that followed. Two clusters were merged together if the distance between centroids of the two was smaller than either σ. The clustered data sets were binary labeled and fed into the GNW for feature ranking. The clusters-labeled data sets were equally divided into a training and test set. The GNW constructed consisted of three layers namely, input, hidden, and output ones. While the number of hidden layer nodes was fixed at 5, the number of input layer nodes was varied with the number of input features. The nodes on each layer were fully connected. The activation function employed for the hidden layer was hyperbolic, while that for the output layer was sigmoid. The biases of each activation function were set at −5.1. The number of nodes on output layer was set as 2 or 3 for 2 or 3 to 8 clusters partitioned. In other words, 2 or 3 binary digits were used to label 2 or 3 to 8 clusters classified.

To proceed with the GA training, each of the connected weights of GNW was represented by a 16 bits binary string and encoded in a chromosome. The range of all the weights was set between −2 and 2. There were certain populations of chromosomes generated within each generation, and two out of them were selected by a roulette wheel[31] mechanism for crossover. Depending on the crossover probability P_c set and a random number R_c generated, the two selected chromosomes underwent crossover if and only if $R_c \leq P_{c,}$ otherwise the pair proceeded without crossover. P_c was set at 0.3 throughout the current studies. R_c was also used to determine a crossover site on the two selected chromosomes. Bits on the left of the crossover site were swapped first, while those on the right were next. The entire population of chromosomes generated was also subjected to a mutation process where every bit in every chromosome was visited and occasionally a *1* is flipped to a *0* bit or vice versa. The probability of mutation P_m was set at 0.1 throughout the current studies. The new population generated was used to replace the old one, and then the corresponding fitness functions were computed. The fitness function was computed as a reciprocal of the sum of the squared difference between the labeled and trained values of each GNW output layer node. To find a global optimum, the fittest (*elite*) chromosome in the entire population was searched and then propagated to the next generation. The number of generation was fixed as 60, while the number of population was varied between 10 and 60. The classification accuracy of the GNW on the training and test set was evaluated by comparing the labeled and trained values of each output layer node.

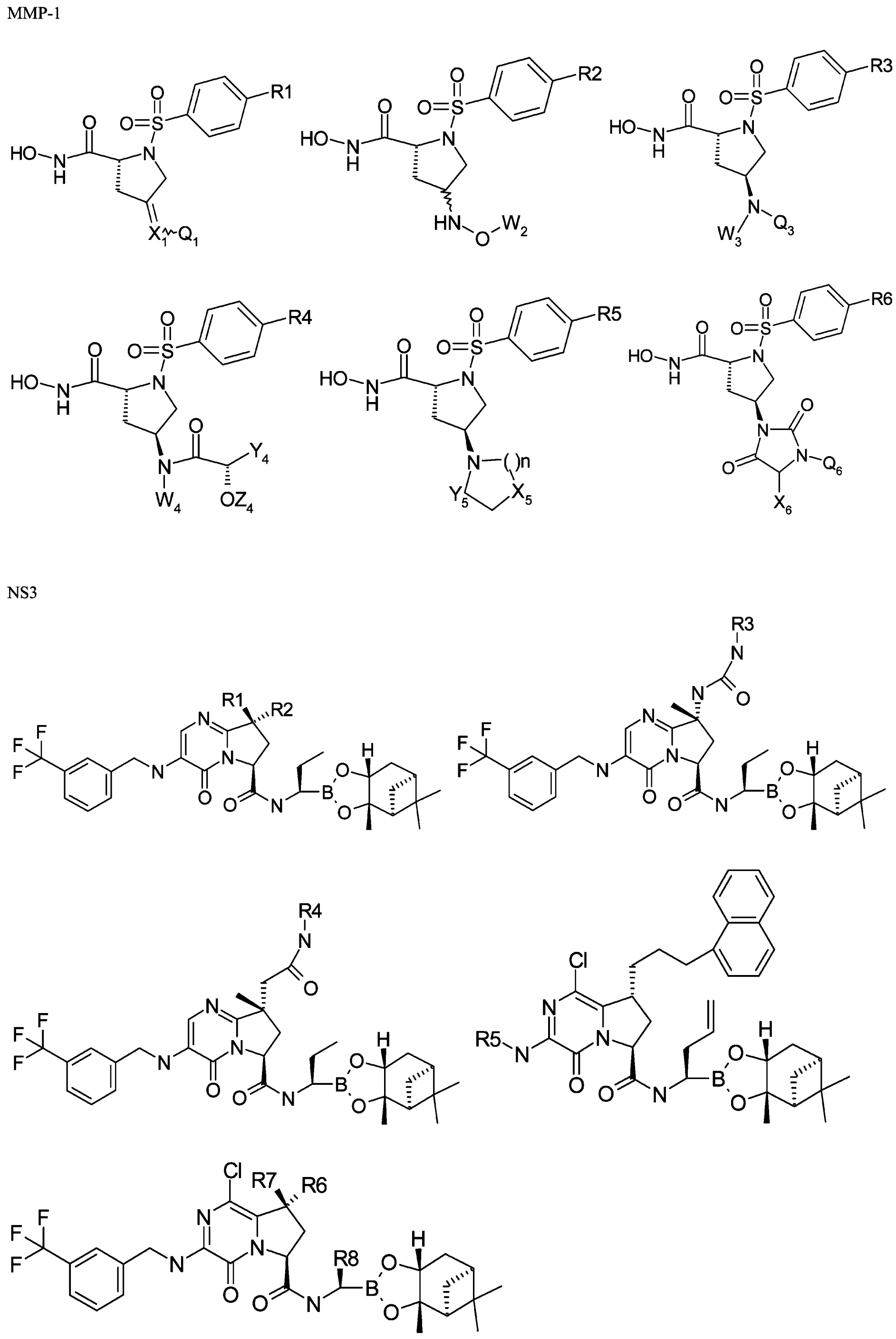

Figure 1. The core structures of the 85 MMP-1 and 62 HCV NS3 protease inhibitors classified.

Each of the input layer nodes along with its outgoing connections with the hidden layer nodes was eliminated once for proceeding with the feature ranking process using the GNW. Suppose that an input layer node h has been removed. Then, the elimination of h involves removing all its outgoing connections and updating the remaining weights incoming into h's projective field P_h in such a way that the net input of every node i of the field remaining approximately unchanged. This amounts to requiring that the following relation holds[19]

$$\sum_{j\in Ri} w_{ij} x_j^m = \sum_{j\in Ri-\{h\}} (w_{ij} + \delta_{ij}) x_j^m \tag{1}$$

for each node i of field P_h and for each training pattern x^m, $m = 1,....., M$. The quantities δ_{ij}'s were the weight adjustments for the weights w_{ij}'s. Simple algebraic manipulations yielded the following linear system:

$$\sum_{j\in Ri-\{h\}} \delta_{ij}\, x_j^m = w_{ih}\, x_h^m \tag{2}$$

Equation 2 was conveniently expressed as a matrix equation of the form $\boldsymbol{A\delta} = \boldsymbol{b}$, which means that it contains MP_h linear equations with $\sum_{i\in P_h}(R_i-1)$ unknown δ_{ij}'s to be solved. The matrix equation was solved using the LU decomposition routines described by Press et al.[32] For each input layer node (feature) i, all the weights except those from the node connected to the hidden layer nodes (the self-depleted weights) were added with the corresponding weight adjust-

Table 1. Molecular Descriptors Computed for the Five Most Active MMP-1 and HCV NS3 Protease Inhibitors Selected

		MMP-1					NS3				
dn[a]	dn[b]	1d7x_19	1d7x_18	1d7x_14	1d7x_16	1d7x_33	1-3j	3i	3y	3e	1-3g
1		8.52	8.52	8.0	7.88	7.82	7.7	7.7	7.4	7.3	7.22
2		158	162	130	188	122	312	324	304	292	268
3		5.35	5.35	4.86	5.82	4.68	7.86	7.9	7.87	7.58	7.57
4		1.24	1.19	1.26	1.4	1.18	0.97	0.93	0.91	0.96	0.96
5		5.62	5.56	5.15	5.53	4.7	7.04	6.98	6.43	6.4	6.53
6		1.63	1.64	1.78	1.65	1.95	1.04	1.12	1.1	1.12	1.04
7		3.06	3.06	3.24	3.06	3.44	2.71	2.57	2.7	2.37	2.19
8		7.08	6.48	5.32	5.44	4.98	6.87	7.13	6.55	6.26	5.56
9		12	112	13	75	7	154	92	137	196	46
10	10′	8	8	8	8	9	15	15	15	0	15
11		19	19	19	19	18	31	31	31	31	0
12		23	23	23	23	11	44	44	44	89	42
13	13′	11	11	0	0	0	30	30	30	30	83
14	14′	14215	14151	11726	16036	11212	26465	25951	26882	24384	24039
15	15′	6.26	5.32	7.54	8.23	9.01	20.47	27.53	10.22	20.29	28.08
16	16′	4.95	5.06	3.95	6.31	3.93	6.35	6.95	6.68	6.21	6.21
17	17′	513	502	421	592	392	988	973	996	907	912
18	18′	442	424	355	482	341	827	813	806	775	759
19	19′	401	396	322	454	306	797	781	794	734	724
20	20′	1.1	1.07	1.1	1.06	1.11	1.04	1.04	1.02	1.06	1.05
21	21′	2134	1922	872	3695	896	6259	7516	6426	5813	5440
22	22′	12	12	11	16	11	15	14	18	14	13
23	23′	10	9	9	10	9	12	12	12	13	12
24	24′	6	6	6	6	7	5	5	7	5	5
25	25′	1.35	1.64	−0.21	1.57	−0.42	12.45	12	10.57	9.67	10.75
	26′						53	53	53	93	46

[a] Descriptor number for MMP-1 inhibitors: 1. pIC_{50}, 2. comparative molecular field analysis, 3. steric using comparative molecular similarity indexes (CoMSIA), 4. electrostatic using CoMSIA, 5. hydrophobic using CoMSIA, 6. donor using CoMSIA, 7. acceptor using CoMSIA, 8. pharmacophore fit using Catalyst, 9. conformation number using Catalyst, 10. hydrogen-bond acceptor using Catalyst, 11. hydrogen-bond donor using Catalyst, 12. hydrophobic using Catalyst, 13. ring aromatic using Catalyst, 14. sum of atomic polarizabilities, 15. dipole moment, 16. radius of gyration, 17. molecular surface area, 18. molecular weight, 19. molecular volume, 20. density, 21. principal moment of inertia, 22. number of rotatable bonds, 23. number of hydrogen-bond acceptors, 24. number of hydrogen-bond acceptors, 25. log of the partition coefficient. [b] Descriptor number for HCV NS3 protease inhibitors: 10′. hydrogen-bond donor using Catalyst, 13′. hydrophobic using Catalyst, 14′. hydrophobic using Catalyst, 15′. hydrophobic using Catalyst, 16′. dipole moment, 17′. radius of gyration, 18′. molecular surface area, 19′. molecular weight, 20′. molecular volume, 21′. density, 22′. principal moment of inertia, 23′. number of rotatable bonds, 24′. number of hydrogen-bond acceptors, 25′. number of hydrogen-bond acceptors, 26′. log of the partition coefficient.

ments δ_{ij}'s computed and then squared and summed and treated as a primary feature ranking index for feature i (frdx_i) defined as follows:

$$\mathrm{frdx}_i = \sum_{l}^{\text{hidden}} \sum_{j\in Ri-\{h\}} (w_{ij}^l + \delta_{ij})^2 \tag{3}$$

To eliminate the effect of clustering on the feature ranking result, each data matrix was clustered numerous times, and the frequency of each feature being ranked in a particular rank was counted. In other words, the features of a data partition j were ranked from bad to good, and the frequency being in each rank frdx_{ij}^l of numerous partitions was counted and weighted by the rank r_{ij}^l of a particular rank l. These weighted frequencies were summed and used as the final feature ranking index for feature i (ffrdx_i) as follows:

$$\mathrm{ffrdx}_i = \sum_{j}^{\text{partitions}} \sum_{l}^{\text{ranks}} \mathrm{frdx}_{ij}^l \cdot r_{ij}^l \tag{4}$$

To monitor the feature selection and pruning process, we have either used the stepwise variable selection program MREG described by Jurs[33] or directly fitted the selected descriptors to pIC_{50} values through a linear relationship by solving a normal equation of the form

$$\beta = (\mathbf{X'X})^{-1}\mathbf{X'y} \tag{5}$$

where β is a ($p \times 1$) vector of the regression coefficients, $\mathbf{X'X}$ is a ($p \times p$) symmetric matrix, $\mathbf{X'y}$ is a ($p \times 1$) column vector, $\mathbf{X}$ is a ($n \times p$) matrix of numbers of the descriptors selected, and $\mathbf{y}$ is a ($n \times 1$) vector of the corresponding pIC_{50} values. The detail of the normal equation could be found elsewhere.[34] The normal equation was solved using the LU decomposition routines described by Press et al.[32] The conventional r^2 defined as follows and F-ratio[35] were used to evaluate the performance of a model generated by pruning off some bad features

$$r^2 = 1 - \frac{\sum(y - \mathrm{pred})^2}{\sum(y - \mathrm{mean})^2} \tag{6}$$

where y was the measured pIC_{50}, and pred and mean were the predicted and mean pIC_{50} values computed from the fitted equation, respectively. The two data matrices were also clustered using the SimpleKMeans method implemented in the WEKA package.[36] Features of the clustered data were ranked according to the F-scores[37] computed which were similar to the Fisher's discriminant ratio[2] described previously.[28] The larger the F-scores computed the greater the discriminative power the features possess. The clustered data were applied to a SVM package LIBSVM[38] for training and predicting. The SVM training with a RBF kernel[38] employed and then scored by the F-scores[37] was also a supervised feature ranking process. The feature ranking indexes for these comparative studies were computed in the

Table 2. GNW Training on an Artificial Data Matrix of Dimension 85 × 24

can[a]	vn[b]	target value			trained (p=10)			trained (p=40)			trained (p=60)		
1	2	0	0	1	0.003	0.404	0.875	0.071	0.396	0.801	0.098	0.570	0.769
1	4	0	0	1	0.003	0.405	0.875	0.071	0.396	0.801	0.098	0.570	0.769
1	6	0	0	1	0.003	0.416	0.878	0.071	0.396	0.801	0.098	0.570	0.769
1	8	0	0	1	0.003	0.407	0.875	0.071	0.396	0.801	0.098	0.570	0.769
1	35	0	0	1	0.003	0.412	0.874	0.072	0.396	0.798	0.099	0.567	0.766
1	37	0	0	1	0.004	0.417	0.837	0.093	0.350	0.746	0.135	0.448	0.679
1	39	0	0	1	0.003	0.415	0.865	0.074	0.388	0.792	0.101	0.559	0.762
1	41	0	0	1	0.003	0.418	0.853	0.095	0.346	0.741	0.142	0.420	0.655
2	43	0	1	0	0.005	0.419	0.773	0.097	0.343	0.736	0.159	0.386	0.625
2	45	0	1	0	0.003	0.417	0.851	0.132	0.293	0.661	0.153	0.400	0.637
2	47	0	1	0	0.010	0.422	0.599	0.101	0.337	0.729	0.123	0.483	0.706
2	49	0	1	0	0.015	0.424	0.465	0.129	0.297	0.667	0.149	0.412	0.647
2	78	0	1	0	0.058	0.433	0.140	0.595	0.085	0.156	0.677	0.024	0.077
2	80	0	1	0	0.058	0.433	0.140	0.595	0.085	0.156	0.677	0.024	0.077
2	82	0	1	0	0.058	0.433	0.140	0.595	0.085	0.156	0.678	0.024	0.077
2	84	0	1	0	0.058	0.433	0.140	0.595	0.085	0.156	0.678	0.024	0.076
3	10	0	1	1	0.003	0.415	0.879	0.071	0.396	0.801	0.098	0.569	0.768
3	12	0	1	1	0.003	0.417	0.879	0.071	0.396	0.801	0.098	0.570	0.769
3	14	0	1	1	0.003	0.410	0.876	0.071	0.396	0.801	0.098	0.570	0.769
3	16	0	1	1	0.003	0.390	0.870	0.071	0.396	0.801	0.098	0.570	0.769
3	18	0	1	1	0.003	0.415	0.878	0.071	0.396	0.801	0.098	0.570	0.769
3	20	0	1	1	0.003	0.415	0.874	0.071	0.396	0.800	0.098	0.569	0.769
3	22	0	1	1	0.003	0.417	0.876	0.071	0.394	0.799	0.098	0.569	0.769
3	24	0	1	1	0.003	0.410	0.876	0.071	0.396	0.801	0.098	0.568	0.768
3	26	0	1	1	0.003	0.416	0.875	0.071	0.396	0.800	0.098	0.568	0.768
3	28	0	1	1	0.003	0.416	0.874	0.071	0.395	0.799	0.099	0.565	0.765
3	30	0	1	1	0.003	0.416	0.875	0.073	0.390	0.794	0.098	0.564	0.765
3	32	0	1	1	0.020	0.427	0.385	0.073	0.390	0.794	0.098	0.565	0.765
4	52	1	0	0	0.037	0.430	0.222	0.436	0.125	0.264	0.556	0.047	0.136
4	54	1	0	0	0.046	0.431	0.179	0.432	0.126	0.268	0.558	0.047	0.135
4	56	1	0	0	0.048	0.431	0.169	0.518	0.103	0.203	0.618	0.034	0.103
4	58	1	0	0	0.051	0.432	0.160	0.505	0.106	0.212	0.576	0.043	0.125
4	60	1	0	0	0.052	0.433	0.156	0.506	0.093	0.176	0.651	0.028	0.088
4	62	1	0	0	0.053	0.432	0.153	0.582	0.088	0.163	0.667	0.026	0.081
4	64	1	0	0	0.055	0.433	0.146	0.579	0.089	0.165	0.662	0.026	0.083
4	66	1	0	0	0.056	0.433	0.144	0.591	0.086	0.158	0.675	0.025	0.078
4	68	1	0	0	0.057	0.433	0.142	0.590	0.087	0.159	0.674	0.025	0.078
4	70	1	0	0	0.057	0.433	0.141	0.593	0.086	0.157	0.676	0.024	0.077
4	72	1	0	0	0.057	0.433	0.141	0.593	0.086	0.157	0.676	0.024	0.077
4	74	1	0	0	0.057	0.433	0.140	0.594	0.086	0.156	0.677	0.024	0.077
4	76	1	0	0	0.057	0.433	0.140	0.595	0.085	0.156	0.677	0.024	0.077

[a] The cluster number assigned for each cluster of the artificial data set classified. [b] The vector number of each member in a cluster of the artificial data set classified.

Table 3. Feature Ranking Indexes frdx$_i$ Computed from the GNW Training on the Artificial Data Set 4ds and 18ds Being Partitioned into 4 to 10 Clusters

K[a]	actual partition	frdx$_i$ ranking (good → bad)
4ds	$4,^{b\ 16c}8,^{19}16,^{22}23^{28}$	
2	4(16,17,25,27)	13,14,17,15,12,18,11,1,3,5,24,22,9,20,10,7,6,21,19,2,**4**,**16**,**8**,**23**
3	6(7,11,12,14,19,22)	7,17,22,19,14,10,20,24,12,5,13,11,18,9,15,6,1,3,21,2,**16**,**23**,**8**,**4**
4	7(7,7,10,11,12,16,19)	13,5,19,1,15,18,20,12,9,24,22,3,14,6,17,10,11,7,21,2,**16**,**4**,**23**,**8**
5	8(6,7,7,8,11,12,15,19)	17,6,18,14,12,5,13,2,9,1,24,15,19,22,10,21,7,3,11,20,**16**,**23**,**8**,**4**
6	9(6,7,7,8,8,11,11,12,15)	21,15,20,9,14,22,5,24,17,13,11,7,3,1,6,19,12,18,2,10,**16**,**8**,**4**,**23**
7	10(5,6,7,7,8,8,10,11,11,12)	15,17,19,20,13,1,14,7,2,24,3,12,5,22,18,21,10,6,11,9,**8**,**16**,**4**,**23**
8	10(5,5,5,6,6,7,8,10,12,18)	5,22,6,17,14,2,7,3,9,13,1,19,12,11,18,24,15,20,10,21,**8**,**4**,**16**,**23**
18ds	$1^3,2^5,3^9,4^{12},6^{14},7^{17},8^{20},9^{20},10^{21},11^{26},13^{25},14^{31},15^{30},16^{31},17^{37},18^{39},19^{40},20^{41}$	
2	4(9,13,29,34)	21,**1**,22,23,12,5,24,**9**,**20**,**11**,**17**,**6**,**2**,**4**,**3**,**7**,**19**,**15**,**18**,**14**,**13**,**16**,**8**,**10**
3	5(4,13,18,20,28)	**1**,21,22,12,23,24,5,**17**,**3**,**18**,**7**,**2**,**13**,**4**,**19**,**9**,**11**,**16**,**6**,**8**,**10**,**20**,**15**,**14**
4	7(4,5,5,14,15,17,25)	21,22,23,2,12,24,5,**11**,**3**,**13**,**1**,**9**,**10**,**20**,**14**,**15**,**4**,**6**,**7**,**17**,**18**,**16**,**8**,**19**
5	8(4,5,6,8,12,15,17,18)	12,24,22,23,5,**1**,21,**3**,**10**,**2**,**4**,**6**,**20**,**7**,**16**,**11**,**18**,**13**,**19**,**8**,**17**,**9**,**14**,**15**
6	8(5,6,8,8,9,12,17,18)	**3**,23,**6**,22,24,12,21,5,**19**,**2**,**10**,**15**,**1**,**13**,**7**,**14**,**11**,**9**,**16**,**20**,**8**,**18**,**17**,**4**
7	9(5,6,6,7,8,8,9,16,17)	23,22,24,12,21,**3**,5,**20**,**2**,**7**,**6**,**13**,**1**,**18**,**8**,**10**,**11**,**9**,**17**,**14**,**15**,**4**,**19**,**16**
8	10(5,5,6,6,6,8,8,9,10,17)	22,5,23,21,24,12,**3**,**20**,**6**,**9**,**2**,**10**,**7**,**17**,**8**,**18**,**1**,**13**,**11**,**15**,**14**,**19**,**4**,**16**

[a] The K value input into the K-means$^+$ clustering method. [b] The feature number selected for changing the descriptor value to 0.1 for the artificial data set. [c] The number of vectors being changed to 0.1 for the feature of the artificial data set.

same way as those described above where the data sets were clustered and trained by the SVM method numerous times, and the frequency of each feature being ranked in a particular rank was counted and summed.

RESULTS AND DISCUSSION

As presented in Figure 1, structures of the 85 MMP-1 inhibitors or 62 HCV NS3 protease inhibitors studied here may be partitioned into 6 or 5 groups, respectively. However, the 24 or 25 molecular descriptors computed for these compounds are extremely diversified as shown in Table 1 for the five most active compounds selected from each inhibitor set. It is difficult to categorize the normalized descriptors using some convenient feature evaluation indices such as entropy,[28] χ^2 statistics,[39] or the Fisher discriminant ratio.[2] To test the feasibility of GNW on ranking these diversified features, we have created some artificial data sets by generating some artificial data sets of random numbers of dimension 85 × 24 (to mimic the MMP-1 inhibitors) and 62 × 25 (to mimic the HCV NS3 protease inhibitors). The magnitude of each column (feature) of these artificial data sets is between 1 and 0 as similar to the normalized data sets. Since all are being created randomly, the goodness of each feature of these artificial data sets is presumably the same. However, a good artificial feature may be changed to a bad one if the descriptor values of some of its corresponding rows (members) are changed to a fixed number such as 0.1. This is monitored as a decrease in conventional r^2 (eq 6) computed for the feature. Therefore, a worse feature is generated by setting more rows of corresponding descriptor values generated as 0.1. The goodness of an artificial feature is then visually judged by how many rows of the feature are being changed to 0.1. For examples, the feature ranking for features A, B, and C would be A > B > C if the numbers of rows being changed to 0.1 for A, B, and C are such that A < B < C.

To test the feasibility of our feature ranking process, these artificial data sets are clustered using the K-means$^+$ clustering scheme and then trained by the GNW. The chosen K values are from 2 to 8, while the actual clusters obtained are from 2 to 12. Some major parameters of the GNW such as the biases and range of weights are determined at the desired levels during some preliminary training runs on the clustered artificial data sets. The important GA parameters such as P_c (number of population) and P_m (number of generation) used in optimizing the GNW training are determined from these preliminary runs. These aforementioned GNW parameters are considered to be effective if they give the same feature ranking results as those seen visually (the known feature rankings). The training by GNW on an artificial data set is presented in Table 2 where only the test set of the artificial data set is shown. The artificial data set has been partitioned into 4 clusters with each being assigned by 8, 8, 12, and 13 members by the K-means$^+$ method, respectively (Table 2). Since using 3 binary digits are sufficient to label these clusters, the number of output layer nodes on the GNW is set as 3, and the corresponding target values are varied from *001* to *100* as shown in Table 2. The partitioned artificial data set is trained with all the important GA parameters fixed except the number of population p is varied from 10 to 60. The trained values of each output layer node for the cases that p is chosen as 10, 40, and 60 are presented in Table 2. The training accuracy of each case is determined by comparing the trained with the target value of each output layer node of each member. A member is considered accurately trained if the trend of its three trained output layer nodes is similar to that of the target ones. For example, member 2 of case $p = 10$ is accurately trained since both the trends of trained ($0.003 < 0.404 < 0.875$) and target ($0 \leq 0 < 1$) agree with each other (Table 2). However, member 52 of the same case is inaccurately trained since the trend of trained ($0.037 < 0.430 > 0.222$) is different from that of target ($1 > 0 \geq 0$) (Table 2). The training accuracy of the GNW on the partitioned artificial data set estimated is 59, 81, and 81% for $p = 10$, 40, and 60, respectively (Table 2). Apparently, there is a certain improvement on the trained values of the output layer nodes for $p = 60$ over that for $p = 40$ (Table 2). The artificial data sets are not overtrained since no further improvement in the trained values of the output layer nodes is observed if p is increased to 70.

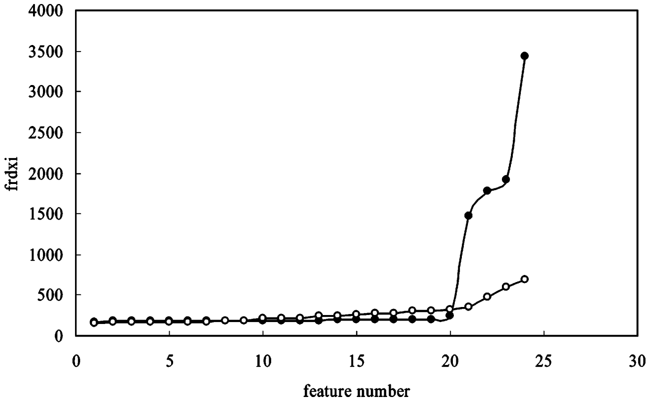

Figure 2. The feature ranking indexes frdx$_i$ computed for the artificial data sets 4ds and 18ds being partitioned into 4 to 10 clusters (see Table 3 for details and see $K = 2$ cases for the rankings of these two plots) are plotted against the feature ranks. Symbols of filled and open circles are used to represent data for 4ds and 18ds data sets, respectively.

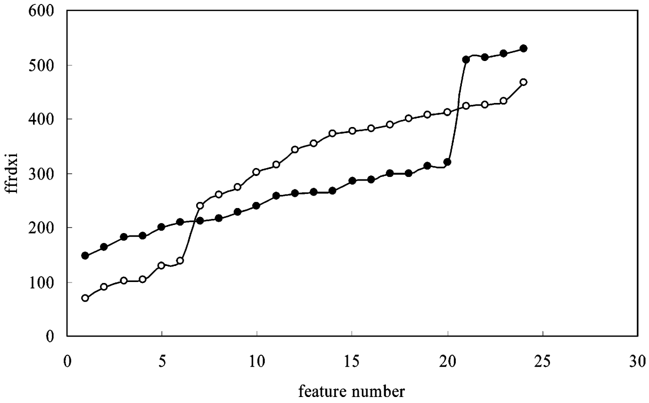

Figure 3. The final feature ranking indexes ffrdx$_i$ computed for the artificial data sets 4ds and 18ds are plotted against the feature ranks. Symbols of filled and open circles are used for representing indexes for 4ds and 18ds data sets, respectively. The feature ranks for 4ds and 18ds are 13(1), 14(2), 17(3), 15(4), 22(5), 3(6), 24(7), 18(8), 7(9), 1(10), 6(11), 21(12), 20(13), 11(14), 12(15), 9(16), 19(17), 10(18), 5(19), 2(20), 8(21), 16(22), 4(23), 23(24), and 22(1), 21(2), 24(3), 5(4), 12(5), 23(6), 2(7), 1(8), 3(9), 4(10), 20(11), 13(12), 9(13), 6(14), 11(15), 7(16), 10(17), 18(18), 17(19), 19(20), 8(21), 15(22), 14(23), 16(24), respectively.

The primary feature rankings according to the frdx$_i$ values determined for two artificial data sets with descriptor values of 4 and 18 features being changed to 0.1 are designated as 4ds and 18ds and are presented in Table 3, respectively. Both the features and descriptors of the members being changed

Table 4. Feature Ranks by ffrdx$_i$ Computed for Some Artificial Data Sets of Fixed Number of Descriptors Classified by the GNW Method

no. descrip	feature ranks							
4	1^{10a}	2^{18}	3^{26}	4^{30}				
	1	2	3	4				
5	1^{8}	2^{19}	3^{22}	4^{34}	5^{39}			
	1	3	2	4	5			
6	1^{10}	2^{18}	3^{24}	4^{34}	5^{38}	6^{40}		
	1	3	2	4	5	6		
7	1^{10}	2^{17}	3^{25}	4^{28}	5^{33}	6^{42}	7^{45}	
	1	3	6	4	2	7	5	
8	1^{7}	2^{17}	3^{24}	4^{30}	5^{38}	6^{41}	7^{49}	8^{52}
	1	2	8	3	4	7	6	5

[a] The number of members whose descriptor values being changed to 0.1 for the artificial data set.

to 0.1 are randomly selected. Although the K values used to cluster these artificial data sets is systematically varied from 2 to 8, the actual clusters obtained by the K-means$^+$ method are from 4 to 10 (Table 3). The bad features for 4ds and 18ds are listed and visually ranked as 4>8>16>23 and 1>2>3>4>6>7>8>9>10>11>13>14>15>16>17>18>19>20, respectively, as according to a gradual increase in the number of members being changed to 0.1 per feature (Table 3). The primary feature rankings by the frdx$_i$ values of the GNW training on each clustered artificial data matrix are also listed in Table 3. Obviously, the primary feature rankings of the artificial data sets of a particular partition are barely in accord with each other except all the bad features (bold digits) are similarly clustered (Table 3). As mentioned above, these bad features are created with certain number of members being changed to 0.1. A clear-cut separation can be seen between the bad and good (light digits) features for 4ds being partitioned into different clusters (Table 3). However, the separation is being fuzzy if more bad features are created such as for 18ds (Table 3). We find that the uncertainty in separation will appear if more than 63% (15/24) of features are changed to bad ones. Nevertheless, this shows that the GNW training is capable of separating the good features from the bad ones.

The primary feature ranking indexes frdx$_i$ computed for the artificial data sets 4ds and 18ds (Table 3) are plotted against the feature rank and presented in Figure 2. The largest frdx$_i$ indexes computed for 4ds and 18ds are 3433 (for feature 23) and 688 (for feature 10), respectively (see Table 3 for the ranks of these features). The rise of these curves shows a clear separation between good and bad features (Figure 2). There are more worse features being generated for 18ds where the worst one is feature 20 with 41 members being changed to 0.1 as compared to feature 23 of 4ds with only 28 members being changed to 0.1 (Table 3). These plots show that within a partition the smaller the number of features being changed to bad ones the greater the frdx$_i$ indexes computed. The final feature ranks determined according to the ffrdx$_i$ indexes computed for these two artificial data sets are plotted against the feature rank and presented in Figure 3. As shown by a certain jump in ffrdx$_i$ indexes computed, the bad features (with larger ffrdx$_i$ indexes computed) are completely separated from the good ones (with smaller ffrdx$_i$ indexes computed) for 4ds and 18ds, respectively (Figure 3). The influence of descriptor size on feature ranking by the GNW training is examined by ranking features for some artificial data sets of a fixed number of descriptors and known feature rankings namely, 4, 5, 6, 7, and 8 features of known feature ranking in each artificial data set studied. As usual, the feature rankings of these artificial features are judged by the number of members whose descriptor values are being changed to 0.1 for each feature (Table 4). Obviously, the GNW ranking given by ffrdx$_i$ indexes computed is exactly the same as the known ranking when the number of features in an artificial data set is only 4 (Table 4). A slight deviation of the GNW from the known ranking can be seen when the number of features in an artificial data set is increased to 5 or 6 (Table 4). A greater difference between the GNW and the known ranking appears when the number of features is increased to 7 or 8 (Table 4). The authenticity of the GNW ranking is deteriorated when the number of features in an artificial data set is enhanced to be greater than 6. However, the GNW ranking offers a clear-

Table 5. Feature Ranks by ffrdx$_i$ Computed for Seven and Three Artificial Data Sets Classified by the GNW and SVM Methods, Respectively

method	feature ranks																						
	4^{17a}	8^{36}	16^{45}	23^{49}																			
GNW	13	14	17	15	22	3	24	18	7	1	6	21	20	11	12	9	19	10	5	2	**8**	**16**	**4**	**23**
SVM	21	18	11	19	15	3	10	9	13	1	5	12	**8**	20	2	**23**	22	14	17	24	**4**	7	**16**	6
	3^{8}	4^{14}	8^{21}	12^{24}	14^{31}	16^{35}	17^{44}																	
GNW	5	18	11	23	21	15	2	24	19	1	20	9	10	7	6	22	13	**4**	**17**	**16**	**3**	**12**	**8**	**14**
SVM																								
	3^{7}	4^{15}	8^{22}	12^{25}	14^{32}	16^{30}	17^{45}	18^{46}	21^{47}	22^{52}														
GNW	5	7	15	23	11	10	20	24	19	1	6	2	9	13	**4**	**3**	**21**	**18**	**12**	**16**	**8**	**17**	**14**	**22**
SVM																								
	1^{7}	4^{12}	5^{19}	7^{27}	9^{28}	11^{33}	12^{40}	18^{41}	20^{46}	22^{51}	23^{45}													
GNW	14	15	17	3	13	21	10	2	19	8	24	16	6	**1**	**4**	**5**	**7**	**9**	**18**	**12**	**22**	**11**	**23**	**20**
SVM	3	6	24	8	2	16	17	**12**	15	**1**	**11**	21	14	**18**	**22**	**9**	19	10	**20**	13	**5**	**23**	**7**	**4**
	1^{6}	3^{12}	5^{14}	7^{24}	8^{25}	9^{30}	11^{34}	13^{37}	14^{38}	16^{46}	18^{45}	20^{45}												
GNW	22	15	23	19	24	10	12	21	6	4	17	2	**1**	**5**	**18**	**3**	**20**	**11**	**8**	**13**	**7**	**9**	**16**	**14**
SVM																								
	2^{3}	3^{6}	5^{9}	7^{12}	8^{15}	10^{18}	11^{18}	12^{21}	14^{24}	15^{25}	16^{25}	17^{29}	18^{32}	19^{34}	20^{37}	22^{36}								
GNW	13	1	4	6	9	21	24	23	**2**	**5**	**7**	**8**	**3**	**11**	**20**	**18**	**10**	**12**	**19**	**16**	**15**	**22**	**14**	**17**
SVM	1	23	24	21	9	6	**20**	**3**	13	**11**	**8**	4	**18**	**12**	**17**	**22**	**7**	**19**	**14**	**2**	**5**	**10**	**15**	**16**
	1^{3}	2^{6}	3^{8}	4^{9}	6^{15}	7^{17}	8^{20}	9^{21}	10^{26}	11^{27}	13^{23}	14^{30}	15^{30}	16^{31}	17^{39}	18^{37}	19^{40}	20^{43}						
GNW	22	21	24	5	12	23	**2**	**1**	**3**	**4**	**20**	**13**	**9**	**6**	**11**	**7**	**10**	**18**	**17**	**19**	**8**	**15**	**14**	**16**
SVM																								

[a] The number of members being converted to 0.1 for the feature selected is represented by a superscript.

cut separation of good from bad features even when the number of features in an artificial data set is increased to 24 as presented in Table 5. In these studies, the number of features in each artificial data set is fixed as 24, while the number of bad features with each being marked with a superscript number to represent the corresponding number of members being changed to 0.1 in each artificial data set is varied from 4 to 18 (Table 5). The bad features in each ranking result identified are represented with bold digits (Table 5). Apparently, the GNW ranking is able to separate all the bad features from the good ones regardless of the number of bad features included in an artificial data set studied (Table 5). In contrast, the SVM ranking is unable to give a similar separation for good from bad features for the three artificial data sets with the number of bad features included being 4, 11, and 16, respectively (Table 5). The separation of good from bad features by the GNW ranking is regarded as 100%, while that by the SVM ranking is rather indistinct. However, for the case of 16 bad features included, the percentage of separation (% separation) for good from bad features by the SVM ranking can be counted as 75 (12/16) % since there are four bad features, namely, 20, 3, 11, and 8 being separated from the other bad ones (Table 5).

The major advantage of GNW ranking is that the effect of data classification on the feature ranking result is eliminated (Tables 3 and 5). Moreover, the GNW ranking is not affected by the training accuracy. The % separation for good from bad features based on ranking the frdx$_i$ values computed for the training and test sets of all the artificial data sets described in Table 5 are plotted against the training accuracy and presented in Figure 4 (parts a and b, respectively). The training accuracy for both the training and test sets is varied from 0.1 to 0.85 (Figure 4a,b). Despite variation in the raining accuracy obtained, the % separation for good from bad features estimated for most of the artificial data sets classified is 100 (Figure 4a,b). For both the training and test sets, the % separation for good from bad features is deteriorated when the number of bad features included in an artificial data set is greater than 11 (Figure 4a,b). The worst % separation for good from bad features is given by the artificial data set with 18 or the most number of bad features included (Figure 4a,b). However, as shown in Table 5, all the % separation for good from bad features equals to 100 for all the artificial data sets classified if the features are ranked using the ffrdx$_i$ instead of frdx$_i$ indexes.

Both the GNW and SVM training are also applied to the classification of some standard data sets such as the Iris,[40] Vowel Recognition,[40] and Ionosphere[40] data sets. The three standard data sets have also been classified by others[41] using the mutual information (MI) driven techniques namely, Supervised Relevance Neural Gas (SRNG) and Energy SRNG (ESRNG). The number of features in the Iris, Vowel Recognition, and Ionosphere data sets is 4, 10, and 34, and the dimensions of these standard data sets are 4 × 150, 10 × 990, and 34 × 351, respectively. As shown in Table 6, both the GNW and SVM trainings give the same feature ranking on the Iris data set. However, both the GNW and SVM trainings barely agree with each other on ranking the features of both the Vowel Recognition and Ionosphere data sets (Table 6). However, the difference between ranking these two latter data sets by the GNW and SVM training appears to be far smaller than that between the former and the two MI based techniques (Table 6). For the Ionosphere data set, only the first five ranks are given by the two MI based techniques[41] (Table 6). These results agree with that presented in Table 4 where the GNW training is shown to give the same ranking as the known ranking on some artificial data sets with less than 5 or 6 features included.

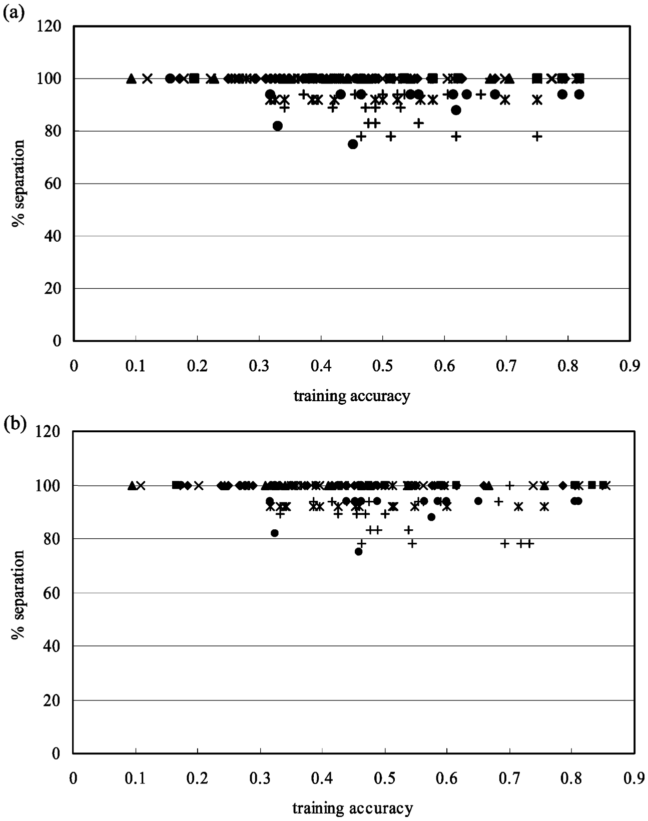

Figure 4. (a) The percentage of separation (%separation) between good and bad features computed by the GNW ranking on several artificial data sets (4ds, 7ds, 10ds, 11ds, 12ds, 16ds, and 18ds) is plotted against the training accuracy for the training sets. The GNW ranking is based on the frdx$_i$ values computed for each data set. Symbols used to represent each data set are designated as follows: filled diamonds for 4ds, filled squares for 7ds, filled triangles for 10ds, multiplies for 11ds, stars for 12ds, filled circles for 16ds, and crosses for 18ds. (b) The percentage of separation (%separation) between good and bad features computed by the GNW ranking on several artificial data sets (4ds, 7ds, 10ds, 11ds, 12ds, 16ds, and 18ds) is plotted against the training accuracy for the test sets. The GNW ranking is based on the frdx$_i$ values computed for each data set. Symbols used to represent each data set are designated as follows: filled diamonds for 4ds, filled squares for 7ds, filled triangles for 10ds, multiplies for 11ds, stars for 12ds, filled circles for 16ds, and crosses for 18ds.

Both the MMP-1 and HCV NS3 protease inhibitor data sets are partitioned into *K* clusters numerous times and then equally divided into the training and test sets and classified by both the GNW and SVM methods. The corresponding training accuracy computed is varied from 0.2 to 0.95 for both the training results. The ffrdx$_i$ indexes computed by both training methods are ranked from good (top) to bad (bottom) for the MMP-1 in Table 7 and HCV NS3 protease inhibitor data set in Table 8, respectively. Apparently, the feature ranks of both data sets by the GNW training are barely in accord with those by the SVM training (Tables 7 and 8). However, features 16, 17, and 18 (molecular surface area, molecular weight, and molecular volume) are unani-

Table 6. Feature Ranks of the Iris, Vowel Recognition, and Ionosphere Data Sets Given by the GNW (ffrdx$_i$), SVM, SRNG,[a] and ESRNG[b] Methods

method	feature ranks																			
Iris	1	2	3	4																
GNW	2	4	1	3																
SVM	2	4	1	3																
SRNG	3	4	2	1																
ESRNG	3	1	4	2																
Vowel R	1	2	3	4	5	6	7	8	9	10										
GNW	9	10	2	8	7	4	1	3	6	5										
SVM	10	8	5	6	4	9	7	1	3	2										
SRNG	1	4	6	2	3	9	8	5	7	10										
ESRNG	2	1	3	8	9	4	5	10	8	7										
Ionosph	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
GNW	2	16	6	18	33	23	4	27	14	5	20	22	10	3	24	32	17	7	8	28
SVM	33	5	15	1	7	17	34	31	3	13	27	32	21	12	19	11	29	10	28	8
SRNG	24	15	12	10	21															
ESRNG	14	8	5	16	3															
Ionosph	21	22	23	24	25	26	27	28	29	30	31	32	33	34						
GNW	26	21	34	19	12	25	9	31	30	11	29	15	13	1						
SVM	14	18	23	30	26	16	9	24	22	25	4	20	6	2						

[a] Feature rank given by the supervised relevance neural gas algorithm described in ref 41. [b] Feature rank given by the energy SRNG algorithm described in ref 41.

Table 7. Feature Rank ffrdx$_i$ Determined for the MMP-1 Inhibitors by the GNW and SVM Trainings Are Listed from Good (Top) to Bad (Bottom)

GNW				SVM			
ffrdx$_i$ rank	r^2	F-ratio	std dev	ffrdx$_i$ rank	r^2	F-ratio	std dev
0(18)	0.785	9.11	0.386	0(16)	0.785	9.11	0.386
16	0.779	9.34	0.389	18	0.785	9.66	0.383
17	0.770	9.45	0.393	17	0.785	10.26	0.380
20	0.766	9.80	0.393	1	0.778	10.53	0.382
1	0.765	10.41	0.391	23	0.776	11.05	0.382
15	0.763	11.02	0.389	4	0.766	11.18	0.387
4	0.687	8.05	0.444	19	0.726	9.69	0.416
2	0.676	8.21	0.449	13	0.725	10.41	0.413
23	0.660	8.25	0.456	20	0.660	8.25	0.456
13	0.650	8.55	0.459	2	0.659	8.89	0.454
7	0.643	8.99	0.461	15	0.591	7.23	0.493
21	0.622	8.99	0.471	21	0.587	7.77	0.492
22	0.617	9.65	0.471	6	0.508	6.20	0.533
10	0.597	9.83	0.479	22	0.448	5.38	0.561
9	0.435	5.71	0.564	3	0.448	5.99	0.558
11	0.431	6.32	0.562	24	0.403	5.63	0.576
6	0.406	6.50	0.570	5	0.403	6.42	0.572
24	0.368	6.39	0.585	11	0.403	7.41	0.568
5	0.288	5.27	0.616	7	0.340	6.71	0.593
12	0.258	5.49	0.625	14	0.265	5.71	0.622
8	0.086	1.89	0.689	8	0.258	6.96	0.621
19	0.081	2.38	0.687	12	0.081	2.38	0.687
14	0.063	2.75	0.689	9	0.063	2.75	0.689
3	0.030	2.58	0.698	10	0.036	3.08	0.696

Table 8. Feature Rank ffrdx$_i$ Determined for the HCV NS3 Protease Inhibitors by the GNW and SVM Trainings Are Listed from Good (Top) to Bad (Bottom)

GNW				SVM			
ffrdx$_i$ rank	r^2	F-ratio	std dev	ffrdx$_i$ rank	r^2	F-ratio	std dev
9(0)	0.720	3.69	0.461	24(0)	0.720	3.69	0.461
21	0.718	3.93	0.456	23	0.720	3.95	0.455
19	0.718	4.21	0.450	5	0.710	4.05	0.456
16	0.718	4.52	0.444	25	0.710	4.33	0.451
22	0.713	4.74	0.442	6	0.704	4.52	0.450
12	0.713	5.08	0.437	18	0.699	4.77	0.447
18	0.712	5.47	0.432	22	0.695	5.03	0.445
15	0.692	5.37	0.442	3	0.695	5.43	0.440
17	0.677	5.43	0.447	19	0.556	3.24	0.525
3	0.677	5.86	0.443	17	0.492	2.72	0.555
10	0.674	6.33	0.440	13	0.476	2.78	0.558
8	0.674	6.92	0.435	1	0.469	2.96	0.555
20	0.436	2.85	0.566	2	0.459	3.12	0.555
7	0.431	3.08	0.563	11	0.456	3.42	0.550
1	0.430	3.43	0.558	21	0.456	3.80	0.545
4	0.397	3.36	0.568	20	0.412	3.56	0.561
2	0.253	1.96	0.626	4	0.410	4.00	0.557
24	0.236	2.04	0.627	7	0.382	4.10	0.564
23	0.203	1.96	0.635	16	0.311	3.48	0.590
25	0.198	2.26	0.631	15	0.304	4.00	0.588
14	0.182	2.49	0.631	10	0.238	3.50	0.609
11	0.137	2.26	0.643	9	0.237	4.43	0.605
6	0.071	1.47	0.661	8	0.139	3.11	0.637
13	0.004	0.11	0.679	12	0.115	3.82	0.640
5	0.001	0.07	0.675	14	0.111	7.47	0.636

mously identified by both methods as the best features among all the 24 ones ranked for the MMP-1 inhibitors (Table 7). To further compare the feature ranks by the GNW or SVM training on each data set, we have gradually omitted one feature at a time according to its feature rank given in Table 7 or 8 for each data set and then computed the following statistics r^2, F-ratio, and std dev (standard deviation of the predicted values) for the predicted pIC$_{50}$ (eq 5), and the results are also presented in Tables 7 and 8 for the MMP-1 and HCV NS3 protease inhibitor data set, respectively. Apparently, the statistics by both GNW and SVM trainings appear to be similar when no feature is deleted from both data sets (Tables 7 and 8). However, deletion of a feature from each data set causes deterioration in statistics (decrease in both r^2 and F-ratio but increase in std dev) for the predicted pIC$_{50}$ in both data sets. Moreover, deletion of a good feature will cause more deterioration in statistics than that of a bad one. More deterioration in statistics can be seen for deletion of features by the GNW training up to feature 2 than by the SVM training up to feature 13 for the MMP-1 inhibitors (Table 7). However, deletion of features for feature 20 to 22 for the same data set by the SVM training causes more deterioration in statistics than that for feature 23 to 10 by the GNW training (Table 7). However, for the same data set, deletion of features for feature 9 to 3 by the GNW training does not cause much deterioration in statistics as compared with that for feature 3 to 10 by the SVM training

(Table 7). These comparisons indicate that feature ranks on the MMP-1 inhibitors given by the GNW training is superior to those given by the SVM training since more features are correctly ranked by the former than by the latter. However, feature ranks given by the SVM training on the HCV NS3 protease inhibitors are somewhat better than those given by the GNW training (Table 8). This is evidenced by more deterioration in statistics for deletion of features for feature 5 to 1 by the SVM training than that for feature 19 to 8 by the GNW training (Table 8). These results are similar to those using the MREG program by adding one variable (feature) at a time according to its feature rank and then computing the resultant statistics r^2, F-ratio, and std dev up to the stage (data not shown here).

CONCLUSION

In this report, we have proposed a method for supervised feature ranking using a genetic algorithm optimized artificial neural network GNW. Unlike the saliency metric conveniently used by a multilayer perceptron in ranking features,[42−44] our method concentrates on removing input nodes[19] and their associated weights with the hidden layer nodes systematically and then adjusting the remaining weights so as to preserve the overall network behavior. The weight adjustments are computed from a system of linear equations, and then they are added with the self-depleted weights to form a basic feature ranking index. All the weights are binary encoded, and they are easily evolved by the GA using simple crossover and mutation operators. The change in training accuracy is monitored during some preliminary runs on some artificial data sets of known feature ranks to determine the best parameter settings for parameters such as the generation and population numbers for evolving the GNW. The basic feature ranking index computed is not affected by the training accuracy of a trained GNW on either the training or test set. We also design a final feature ranking index which is independent of the fact how a data set is partitioned. The feasibility of the feature ranking scheme is tested using some artificially generated as well as standard data sets. The feature ranking results on three artificial and standard as well as the MMP-1 and HCV NS3 protease inhibitor data sets given by the GNW are compared with those given by the SVM method. The GNW outperforms the SVM method either in ranking features or separating good from bad features in all three artificial data sets tested. However, both the GNW and SVM methods give the same ranking on a standard data set where only four features are included in the data set. To compare the feature ranking on both the MMP-1 and HCV NS3 protease inhibitor sets by both the GNW and SVM methods, we systematically drop a feature out from the complete pool or add a feature at a time to a smaller pool formed by the three best features ranked and selected for each data set. We find that while the GNW method slightly outperforms the SVM one on one data set, a better performance on the other one is given by the latter. This reveals that feature ranking is a complex task and one often needs to use more than one method in order to reveal various properties of the multivariate data sets. The GNW method presented here will be useful to many pattern recognition problems since the method can give a clear-cut separation of good from bad features when the bad ones in a data set are intrinsically less populated than the good ones.

ACKNOWLEDGMENT

This work is supported in part by a grant from the National Science Council, ROC (NSC94-2313-B007-001). The SYBYL and Catalyst computations are conducted at the National Center for High Performance Computing, ROC.

REFERENCES AND NOTES

(1) Devijver, P. A.; Kittler, J. *Pattern Recognition: A statistical approach*; Prentice Hall: Englewood Cliffs, 1982.

(2) Theodoridis, S.; Koutroumbas, K. *Pattern Recognition*; Academic Press: New York, 1998.

(3) Pudil, P.; Novovicova, J.; Kittler, J. Floating search methods in feature selection. *Pattern Recognit. Lett*. **1994**, *15*, 1119−1125.

(4) Aha, D. W.; Bankert, R. L. A comparative evaluation of sequential feature selection algorithms. In *Artificial Intelligence and Statistics*; Fisher, V. D., Lenz, J. H., Eds.; Springer Verlag: New York, 1996.

(5) Pal, S. K.; Wang, P. P. *Genetic Algorithms for Pattern Recognition*; CRC Press: Boca Raton, FL, 1996.

(6) Kudo, M.; Sklansky, J. Comparison of algorithms that selects features for pattern classifiers. *Pattern Recognit.* **2000**, *33*, 25−41.

(7) Skalak, D. Prototype and feature selection by sampling and random mutation hill climbing algorithms. *Proceedings of the 11th International Machine Learning Conference*, 1994; pp 293−301.

(8) Moore, A. W.; Lee, M. S. Efficient algorithms for minimizing cross validation error. *Proceedings of the 11th International Machine Learning Conference*, 1994.

(9) Liu, H.; Setiono, R. Some issues in scalable feature selection. *Expert Syst. Appl.* **1998**, *15*, 333−3339.

(10) Richard, M. D.; Lippmann, R. P. Neural network classifiers estimate Bayesian a posteriori probabilities. *Neural Comput*. **1991**, *3*, 461−483.

(11) Siedlecki, W.; Sklansky, J. On automatic feature selection. *Int. Natl. J. Pattern Recognit. Artif. Intelligence* **1988**, *2*, 197−220.

(12) Reed, R. Pruning algorithms − a survey. *IEEE Trans. Neural Networks* **1993**, *5*, 740−747.

(13) Cibas, T.; Soulié, F. F.; Gallinari, P.; Raudys, S. Variable selection with optimal cell damage. *Proceedings of the International Conference on Artificial Neural Networks*, Sorrento, Italy, 1994; pp 727−730.

(14) Karnin, E. D. A simple procedure for pruning back-propagation trained neural network. *IEEE Trans. Neural Networks* **1990**, *1*, 239−242.

(15) Le Cun, Y. et al. Optimal brain damage. In *Neural Information Processing Systems II*; Touretzky, D. S., Ed.; Morgan Kaufmann: San Mateo, CA, 1990; pp 598−605.

(16) Mao, J.; Mohiudden, K.; Jain, A. K. Parsimonious network design and feature selection through node pruning. *Proceedings of the 12th International Conference on Pattern Recognition. Jerusalem*, 1994; pp 622−624.

(17) Mozer, M. C.; Smolensky, P. Skeletonization: A technique for trimming the fat from a network via relevance assignment. In *Advances in Neural Information Processing Systems I*; Touretzky, D. S., Ed.; Morgan Kaufmann, San Mateo, CA, 1990.

(18) Stepps, J. M.; Bauer, K. W. Improved feature screening in feed forward neural networks. *Neurocomputing* **1996**, *13*, 47−58.

(19) Castellano, G.; Fanelli, A. M. Variable selection using neural-network models. *Neurocomputing* **2000**, *31*, 1−13.

(20) Cheng, M.; De, B.; Almstead, N. G.; Pikul, S.; Dowty, M. E.; Dietsch, C. R.; Dunaway, C. M.; Gu, F.; Hsieh, L. C.; Janusz, M. J.; Taiwo, Y. O.; Natchu, M. G.; Huddicky, T.; Mandel, M. Design, synthesis, and biological evaluation of matrix metalloproteinase inhibitors derived from a modified praline scaffold. *J. Med. Chem.* **1999**, *42*, 5426−5426.

(21) Glunz, P. W.; Douty, B. D.; Decicco, C. P. Design and synthesis of bicyclic pyrimidinone-based HCV NS3 protease inhibitors. *Bioorg. Med. Chem. Lett.* **2003**, *13*, 785−788.

(22) Zhang, X.; Schmitt, A. C.; Jiang, W.; Wasserman, Z.; Decicco, C. P. Design and synthesis of potent, non-peptide inhibitors of HCV NS3 protease. *Bioorg. Med. Chem. Lett.* **2003**, *13*, 1157−1160.

(23) Huang, H.; Zhang, R.; Xiang, F.; Makedon, F.; Shen, L.; Hettleman, B.; Pearlman, J. K-mean$^+$ method for improving gene selection for classification of microarray data. *IEEE Computational Systems Bioinformatics Conference-Workshops*, 2005; pp 110−111.

(24) Boser, B.; Guyon, I.; Vapnik, V. A training algorithm for optimal margin classifiers. In *Proceedings of the fifth Annual Workshop on Computational Theory*, 1992.

(25) Cortes, C.; Vapnik, V. Support-vector network. *Machine Learning* **1995**, *20*, 273−297.

(26) Witten, I. H.; Frank, E. *Data Mining: Practical machine learning tools and techniques*, 2nd ed.; Morgan Kaufmann: San Francisco, CA, 2005.

(27) Burges, C. J. C. A tutorial on support vector machines for pattern recognition. *Data Min. Knowledge Discovery* **1998**, *2*, 121−167.

(28) Lin, T. H.; Li, H. T.; Tsai, K. C. Implementing the Fisher's discriminant ratio in a K-means clustering algorithm for feature selection and data set trimming. *J. Chem. Inf. Comput. Sci.* **2004**, *44*, 76−87.

(29) *SYBYL 7.0*; The Tripos Associates: 1699 S. Hanley Rd., St. Louis, MO.

(30) *Catalyst 4.7*; The Accelrys Software Inc.: San Diego, CA.

(31) Goldberg, D. E.; Deb, K. A comparative analysis of selection schemes used in genetic algorithms. In *Foundations of Genetic Algorithms*; Rawlins, G., Ed.; Morgan Kaufmann: 1991; pp 69−93.

(32) Press, W. H.; Flannery, B. P.; Teukolsky, S. A.; Vetterling, W. T. *Neumerical Recipes: The Art of Scientific Computing*; Cambridge University Press: New York, 1986.

(33) Jurs, P. C. *Computer Software Applications in Chemistry*; John Wiley & Sons: New York, 1996.

(34) Montgomery, D. C.; Peck, E. A. *Introduction to linear regression analysis*; John Wiley & Sons: New York, 1982.

(35) Golbraikh, A.; Tropsha, A. Beware of q^2! *J. Mol. Graphics Modell.* **2002**, *20*, 269−276.

(36) The WEKA Web site: http://www.weka.net.nz/

(37) Kohavi, R.; John, G. Wrappers for feature subset selection. *Artif. Intelligence J.* **1997**, *1*, 273−324.

(38) Chang, C. C.; Lin, C. J. LIBSVM: a library for support vector machines, 2001. Software available at http://www.csie.ntu.edu.tw/∼cjlin/libsvm.

(39) Liu, H.; Li, J.; Wong, L. A comparative study on feature selection and classification methods using gene expression profiles and proteomic patterns. *Genome Informatics* **2002**, *13*, 51−60.

(40) The Iris, Vowel Recognition, and Ionosphere data sets are available at http://www.grappa.univ-lille3.fr/∼torre/guide.php.

(41) Andonie, R.; Cataron, A. Feature ranking using supervised neural gas and information energy. In *Proceedings of the International Joint Conference on Neural Networks*, Montreal, Canada, July 31, 2005, pp 1269−1273.

(42) Belue, L. M.; Bauer, K. W., Jr. Determining input features for multilayer perceptrons. *Neurocomputing* **1995**, *7*, 111−121.

(43) Ruck, D. W.; Rogers, S. K.; Kabrisky, M. Feature selection using a multilayer perceptron. *Neural Network Comput*. **1990**, *20*, 40−48.

(44) Karnin, E. D. A simple procedure for pruning back-propagation trained neural networks. *IEEE Trans. Neural Networks* **1990**, *1*, 239−242.

CI0600354