VERSITA

Biologia **64**/4: 649—654, 2009
Section Cellular and Molecular Biology
DOI: 10.2478/s11756-009-0125-4

A hybrid genetic-neural model for predicting protein structural classes

Samad Jahandideh[1,2], Somayyeh Hoseini[3], Mina Jahandideh[4] & Mohammad Reza Davoodi[5]

[1]*Department of Biophysics, Faculty of Science, Tarbiat Modares University, P.O. Box 14115/175, Tehran, Iran; e-mail: jahandideh@modares.ac.ir*
[2]*Department of Medical Physics, Shiraz University of Medical Sciences, Shiraz, Iran*
[3]*Department of Biochemistry, Division of Genetics, Tabriz University of Medical Sciences, Tabriz, Iran*
[4]*Department of Mathematics, Faculty of Science, Vali-E-Asr University, Rafsanjan, Iran*
[5]*Department of Electrical Engineering, Faculty of Engineering, Tarbiat Modares University, Tehran, Iran*

Abstract: A genetic algorithm (GA) for feature selection in conjunction with neural network was applied to predict protein structural classes based on single amino acid and all dipeptide composition frequencies. These sequence parameters were encoded as input features for a GA in feature selection procedure and classified with a three-layered neural network to predict protein structural classes. The system was established through optimization of the classification performance of neural network which was used as evaluation function. In this study, self-consistency and jackknife tests on a database containing 498 proteins were used to verify the performance of this hybrid method, and were compared with some of prior works. The adoption of a hybrid model, which encompasses genetic and neural technologies, demonstrated to be a promising approach in the task of protein structural class prediction.

Key words: genetic algorithm; artificial neural networks; sequence parameters; amino acid composition.

Abbreviations: ANNs, artificial neural networks; EF, evaluation function; GA, genetic algorithm; LDA, linear discriminant analysis; MCC, Matthews correlation coefficient; MLR, multinomial logistic regression; PA, prediction accuracy; SR, success rate (sensitivity); 3D, three-dimensional.

Introduction

The functional properties of proteins are depending on their three-dimensional (3D) structure which is encoded in the amino acid sequence. All information regarding the structure and function of a protein is thus coded in its amino acid sequence (Anfinsen 1973). Understanding the rules by which 3D structures of proteins are developed from their linear sequences, is of great importance in contemporary molecular biology. It is also helpful in predicting the function of novel designed sequences.

As far back as 50 years ago it was demonstrated that in some cases the bare protein sequence information is both a necessary and sufficient determinant for the structure and functionality of a peptide, and this paradigm has held true to this day for all but a miniscule minority exceptions to the rule (Anfinsen 1973).

Structural protein classes were defined over 20 years ago as being general ways of describing folds that reflected content of the secondary-structure elements and their arrangement in folded proteins (Levitt 1976). Protein folds can be classified into four main classes consist of all-α, all-β, α/β and $\alpha+\beta$. Since then, various quantitative classification rules have been proposed based on the percentages of α-helices and β-sheets in a protein.

Historically, Nishikawa's findings (Nishikawa & Ooi 1982; Nishikawa et al. 1983a,b) highlighted the strong correlation between the structural classes of proteins and amino acid composition. Since then, many different theoretical methods have been proposed to predict the structural class of proteins, such as statistical analysis which uses parameters obtained from known protein sequences and tertiary structure (Chou & Fasman 1974), information theory (Garnier et al. 1978), nearest neighbor methods (Yi & Lander 1993), multiple alignment (Russell & Barton 1993; King & Sternberg 1996;), neural networks (Metfessel et al. 1993), component-coupled (Chou & Maggiora 1998), combination of multiple alignment and neural networks (Rost & Sander 1994), 3D-one-dimensional compatibility (Ito et al. 1997), support vector machines (Cai et al. 2001), rough sets (Cao et al. 2006), pseudo amino acid composition (Xiao et al. 2008a,b) and hybrid models (Jahandideh et al. 2007a,b). In addition, many more studies have applied various methods to predict protein structural classes (Chou 1995, 1999, 2000, 2005; Chou &

Zhang 1994; Feng et al. 2005; Shenet et al. 2005; Cao et al. 2006; Chen et al. 2006a,b; Du et al. 2006; Niu et al. 2006; Xiao et al. 2006).

In our previous works (Jahandideh et al. 2007a,b), we established hybrid models using multinomial logistic regression (MLR) and linear discriminant analysis (LDA) in the first stages of hybrid modelling procedures and artificial neural networks (ANNs) in the second stages. Results of previous works showed that combination of ANNs as a non-algorithmic model and MLR and LDA both as algorithmic models provide better results than either one alone. At the present study, we applied the neural network analysis to the database for predicting the protein structural classes based on parameters that had been extracted from the protein sequences and automatically selected by a genetic algorithm (GA). Indeed, GA replaced MLR and LDA as a non-algorithmic model in new hybrid modelling procedure.

Material and methods

Database

The database comprising 498 protein domain sequences as described by Zhou (1998) which was collected from SCOP database (Murzin et al. 1995) was used in the present study. The unit of classification in SCOP is usually the protein domain. Small proteins, and most of those of medium size, have a single domain and are therefore treated as a whole. The domains in large proteins are usually classified individually. According to the SCOP, the classification of structural classes for protein domains is based on the evolutionary relationship and on the principles that govern the 3D structure of proteins, therefore is more natural and reliable. We used the database to test our model through self-consistency test and jackknife test and to compare the prediction accuracy and success rates of each structural class with other models.

Our parameters including amino acid and all dipeptide composition frequencies were generated using in-house programs in MATLAB language. Amino acid and dipeptide compositions have been used for several applications, such as predicting protein subcellular localization, virulent proteins in bacterial pathogens, protein secondary structure content, etc. (Liu & Chou 1999; Garg & Gupta 2008; Tantoso & Li 2008). In order to check the fidelity of these programs, results were compared with the outputs of COMPSEQ program (http://bioweb.pasteur.fr/seqanal/interfaces/compseq.html) on the same database.

Feature selection

Feature selection is one of the most important steps in classifier design, because the presence of ineffective features often degrades the performance of a classifier on test samples (Chan et al. 1998). In this article, we used the GA method for feature selection. The GA has become increasingly popular in feature selection as an optimization task (Goldberg 1989). The fundamental principle underlying GAs is the mimicry of natural selection. To solve an optimization task, a GA generates a population of bit strings, which are referred to as chromosomes. Each "chromosome" in that population corresponds to a possible solution of the problem (Qian et al. 2005). In this study, we extracted 420 parameters to train the ANNs. Therefore, there are 2^{420} total possible feature subsets for GA to select the best ones. A binary vector in a 420-dimension space represents an individual in the population. Therefore, the defined chromosome contains 420 genes, one gene for each feature, which takes on 2 values. A value of 0 indicates that the corresponding feature is not selected, and a value of 1 means that the feature is selected. In each generation, the population is probabilistically modified, generating new chromosomes that may have a better chance of solving the problem (Zhang et al. 2005). New characteristics are introduced into a chromosome by crossover and mutation. The probability of survival or reproduction of an individual depends more or less on its fitness to the environment. Each feature in a given feature space is treated as a gene and is encoded by a binary digit (bit) in a chromosome (Wang 2005). In this article, two-point binary crossover and binary mutation are performed.

Selection of individuals to produce successive generations plays an extremely important role in a GA. Selection means that two individuals from the whole population of individuals are selected as 'parents', and the selection is dependent on the individual's evaluation function of each individual. There are several selection schemes. Here the roulette wheel selection was used. This selection simulates a roulette wheel with the area of each segment proportional to its expectation. The algorithm then uses a random number to select one of the sections with a probability equal to its area.

The relevant parameter settings which we used were: population size: 30; number of generation: 100; probability of crossover: 0.8; probability of uniform mutation: 0.1. The size of population is one of the most important parameters. Setting the population size too small may yield premature convergence of GA, while setting the large size of population remains the population variety that could enable GA to search more point and thereby prevent local optimum trapping of the algorithms. However, the time used for a population improvement might be too long for the large population size. We used the equation (1) as evaluation function (EF) of the model and our goal was to maximizing it in test cases:

$$\mathrm{EF} = (p(c) + n(c))/t \times 100 \quad (1)$$

This value was regarded as a measure of fitness in the corresponding generation. This function will be described in performance measures section. For the simulation of this hybrid model, the parameters were selected using GA optimization method and normalized between 0 and 1 according to the maximum value of each feature in the database. The normalized data was then fed forward into the network. This procedure is shown in Figure 1.

The iterative algorithm to evolve a solution to a problem on a computer is including four steps that could be summarized as follows: (i) population preparation; (ii) fitness evaluation; (iii) selection; and (iv) crossover and mutation. Because of their simple and straightforward mechanism as mentioned earlier, GA can be seen as a tool empowering researchers' competence in scientific investigation.

Neural classifier

A neural network is a model that simulates the functions of biologic neurons. The ability of a single neuron could greatly be improved via connection of multiple neurons in a layer. ANNs are powerful non-linear models used vastly for classifying different types of data. A neural network is composed of few layers of neurons. Neurons in adjusting layers are connected with relative quantitative weights. These weights are randomly chosen, and then are changed through the training procedure, so that the mean of the

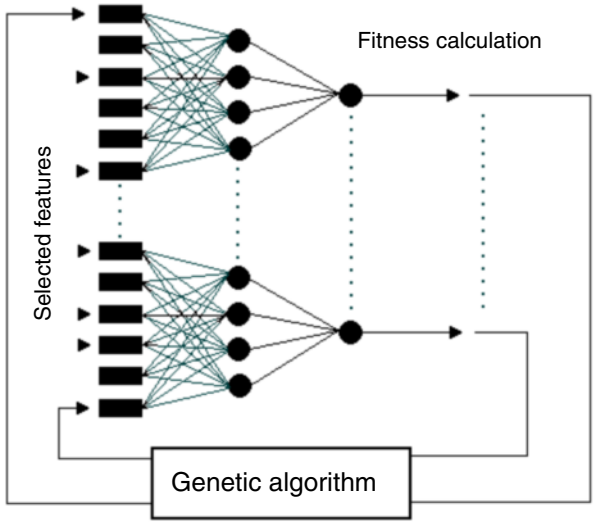

Fig. 1. Feature selection based on neural network classification.

sum-of-squares error is minimized. The minimized sum-of-squares error is the squared difference between the network output and network target, averaged over all of the cases. We used a three-layered feed-forward neural network. Some training algorithms were implemented and tested: gradient descent methods, resilient back-propagation, conjugate gradient methods, and quasi Newton methods. The best results were obtained using a conjugate gradient method. The selected features from GA were considered as inputs into the established neural network. The number of the inputs is automatically optimized using GA processing. The numbers of nodes in the only hidden layer were adjusted in an attempt to achieve optimum classification accuracy on the testing cases. Two nodes were used in the output layer which have been trained to represent [1 1] for all-α, [0 1] for all-β, [0 0] for $\alpha + \beta$ and [1 0] for α/β. Each neuron in the network used a logistic activation function. In order to determine the best optimized structure for the neural network, we simulated a large number of neural networks by varying the number of hidden nodes, iterations and learning rates. Finally, after the network had been trained perfectly in each simulation the testing cases were presented to the trained network. Our network was trained perfectly over 1,000 iterations in each learning process on a personal computer (Pentium 2.8 MHz, IBM compatible machine). Also the optimal learning rate and error goal was found to be 0.2 and 0.02, respectively. The jackknife technique, in which all cases were undergone both the training and testing processes, was applied to train and test model on the database. The procedure is as follows: given a training set of N proteins, the first protein in the training set, t_1, is set aside (left out). Then the model is trained on the remaining $N-1$ proteins and tested on the left out sample. Then sample t_1 is inserted back into the database and the next protein, t_2, is left out. This procedure is repeated until every protein in the database had the opportunity to be a left out sample. It therefore provided as many simulations as the number of samples in each database. In addition to jackknife test we used self-consistency test on the database.

Performance measures

Three threshold dependent indices used to assess the performance of the model can be derived from the four scalar quantities: (i) $p(c)$ is the number of properly predicted proteins in class c; (ii) $n(c)$ is the number of correctly predicted proteins not in class c; (iii) $u(c)$ is the number of under-predicted and $o(c)$ is the number of over-predicted proteins. These indices were used to calculate the prediction accuracy (PA), success rate (sensitivity) (SR) and Matthews correlation coefficient (MCC) for the output of the hybrid model.

1. PA is the total number of correctly classified examples: PA $= (p(c) + n(c))/t \times 100$, where t stands for the number of examples.

2. SR is the percentage of correctly predicted examples in each class: SR $= (p(c)/\mathrm{Ind}(c)) \times 100$, where $\mathrm{Ind}(c)$ is the number of proteins reside in class c.

3. MCC – we used MCC as a more vigorous measure to evaluate the reliability of the established method (Mathew 1975). The MCC for each class is defined by

$$\mathrm{MCC}(c) = (p(c)n(c) - u(c)o(c))/$$
$$/\sqrt{(p(c) + u(c))(p(c) + o(c))(n(c) + u(c))(n(c) + o(c))}$$

The MCC is a limited number between -1 and 1. If there is no relationship between the predicted values and the actual values, the MCC should be 0 or very low (the predicted values are not better than random numbers). In contrast, the MCC value would increase as the strength of the relationship between the predicted values and actual values increases. It is obvious that a perfect fit gives a coefficient of 1.0. Furthermore, the higher MCC indicates the better performance of the prediction for the model.

Results

Many experiments were carried out to find the combination of features that provide the best accuracy in prediction of protein structural classes. The PA on testing samples was used as a measure to calculate the fitness for reproduction of genetic feature selection. We changed the number of hidden units from 3 to 15 and run GA for each of them to show which parameters are more frequently selected. Using the selected parameters, the maximum accuracy was met 93.98% using 13 hidden units in jackknife procedure. In each jackknife process different parameters were selected by the model. However, the only 11 parameters including valine amino acid composition frequency and cysteine-arginine, alanine-glycine, aspartic acid-cysteine, glutamic acid-tyrosine, serine-tryptophan, praline-cysteine, glycine-glutamic acid, histidine-tyrosine, leucine-aspartic acid and tryptophan-asparagine dipeptide composition frequencies were selected in all jackknife processes. All of eleven sequence parameters selected by this model are among selected sequence parameters in our previous works using two algorithmic at the first stages of our hybrid models. Using the same number of hidden units, we ran GA through self-consistency test.

The results of jackknife and self-consistency tests were evaluated by the performance evaluative measures. The results shown in Table 1 were obtained according to the output of the model. Results show that the SR has the highest value in all-α protein structural class. These results are in agreement with our results from previous works (Jahandideh et al. 2007a,b).

Table 1. Performance comparison between self-consistency and jackknife.

Test	Performance measures	Rate of correct prediction for each class			
		All-α	All-β	α/β	$\alpha+\beta$
Self-consistency	SR (%)	100	100	100	100
	MCC	1	1	1	1
Jackknife	SR (%)	96.26	92.06	94.12	93.08
	MCC	0.73	0.75	0.67	0.69

Table 2. Results of self-consistency and jackknife tests.

Test	Algorithm	Success rate (sensitivity) for each class (%)				Predicition accuracy (%)
		All-α	All-β	α/β	$\alpha+\beta$	
Self-consistency	Component coupled[a]	95.80	95.20	94.90	95.40	95.80
	Neural network[b]	100	98.40	96.30	84.50	94.60
	SVM[c]	100	100	100	100	100
	Rough sts[d]	100	100	100	100	100
	Multinomial logistic regression[e]	100	100	100	100	100
	Hybrid neural logistic model[e]	100	100	100	100	100
	LDA[e]	100	100	100	100	100
	Hybrid neural discrimnant model[e]	100	100	100	100	100
	Hybrid genetic neural model	100	100	100	100	100
Jackknife	Component coupled[a]	93.50	88.90	90.40	84.50	89.20
	Neural network[b]	86	96	88.20	86	89.20
	SVM[c]	88.80	95.20	96.30	91.50	93.20
	Rough Sets[d]	87.90	91.30	97.10	86	90.80
	Multinomial logistic regression[e]	92.50	88.10	90.50	89.90	90.40
	Hybrid neural logistic model[e]	96.30	92.10	95.60	93.80	94.40
	LDA[e]	94.39	89.68	92.64	92.24	92.17
	Hybrid neural discrimnant model[e]	95.32	88.88	94.11	93.02	92.77
	Hybrid genetic-neural model	96.26	92.06	94.12	93.08	93.98

[a]Reported results from Zhou (1998). [b]Reported results from Cai & Zhou (2000). [c]Reported results from Cai et al. (2001). [d]Reported results from Cao et al. (2006). [e]The results of these models were reported from our previous works (Jahandideh 2007a,b).

PA and SR in each class for the hybrid genetic-neural model in comparison with our previous hybrid models and some of other methods on the same database are shown in Table 2. The results of the hybrid genetic-neural model showed all the percentages of correct prediction on the database reaching 100% in self-consistency test, which is the same as our previous hybrid models results, SVM and rough sets based methods (Cai et al. 2001; Cao et al. 2006; Jahandideh et al. 2007b). The highest PA value of 94.6% has been obtained in a previous study using jackknife test (Feng et al. 2005). However, our results indicated that the hybrid genetic-neural model, same as our previously proposed hybrid models, captured the characteristics between sequences and their classes through single amino acid and all dipeptide composition frequencies. The comparison should be focused on the jackknife test because it is more rigorous and objective method. From the result of jackknife test, it is obvious that the PA is comparable to previous proposed models.

Discussion and conclusion

Determining the 3D fold of a protein applying golden standard techniques such as NMR and X-ray crystallography is expensive and time-consuming. Consequently, there is a large gap between the number of known protein sequences and the number of known 3D protein structures. The computational prediction of structures from amino acid sequence has therefore come to play a key role in narrowing the gap. The previous reports indicated that these computational methods have been very promising in providing useful information for the biological research community.

In order to establish powerful hybrid models we used GA in conjunction with neural network in the present work. In the previous works, MLR and LDA were used at the first stage of hybrid modelling procedures. Indeed, GA, LDA and LDA were used to select the effective sequence parameters that are applied for prediction of protein structural classes.

We note that while the accuracy of predicting four major classes of proteins appears promising, SCOP does not includ only four classes. Currently, there are eleven classes listed, of which at least seven are highly populated (Murzin et al. 1995). Also these proteins and folds are further classified into superfamilies, families, etc. Any classification technique should attempt to predict the sub-classes as well, in addition to the main classes. For instance, Rogen & Fain (2003) successfully

reproduced the entire hierarchy of CATH classification database (Orengo et al. 1997) (∼95% accuracy). The attempts then should be made to produce similar results for SCOP, rather than just predict only the major classes at the first level of hierarchy.

In our opinion the high accuracies obtained by majority of existing methods on dataset 498 proteins are an artifact resulting from duplicates and highly similar sequences included in this dataset. Our dataset with 498 domains include over 10 copies of the same sequence that appears under different PDB (http://www.rcsb.org/) codes. The accuracies of the same models on the databases with low sequence identities are lower (Chen et al. 2007).

Regarding the fact that almost all previously used models detected all-α cases better than other classes (Gromiha & Selvaraj 1998), it is revealed that the SR value in the result of hybrid genetic-neural model is compatible many other previous works done. A plausible reason for this tendency of predictors is the predominant role of short and medium range interactions in all-α proteins. Similarly, uniformly lower accuracy in the prediction of the other classes implies the dominance of long-range interactions (Gromiha & Selvaraj 1998).

It is obvious from the results of jackknife test, α/β class has the higher SR than has the $\alpha + \beta$ class. This may be related to the proportion of the α/β class in the training sets in which α/β class occupied the bigger part. As a supervised learning method, it makes it easier to capture characteristics that feed more training objects to neural networks.

Using restricted number of sequence parameters among the 420 sequence parameters, hybrid models can predict the structural class of proteins; this is one of the most important advantages of these models. In general, the results showed that by use of hybrid genetic-neural model, one can provide adequate information for an accurate prediction applying a few sequence parameters, only including single and dipeptide compositions. Although the ANNs may work as an excellent predictor, it may not be able to explain which findings are more relevant in reaching the pattern recognition due to its "black box" behaviour (Randall et al. 2006). GA offers a particularly attractive approach to select the effective parameters (Zhang et al. 2005). According to the obtained results, applying GA improved the PA. The benefits of applying GA as a pre-processor were improvement in generalization ability of ANNs and reducing the size of calculations through simplifying the ANNs structure.

This study clarified the efficiency of using hybrid genetic-neural model in determining effective parameters, as well as an independent predictor. Moreover, the optimal structure of neural network can be simplified, thereby reducing the needed time for neural network training procedure and the probability of over-fitting occurrence is decreased and a high precision and reliability is obtained in this way.

References

Anfinsen C.B. 1973. Principles that govern the folding of protein chains. Science **181:** 223–230.

Cai Y.D., Liu X.J., Xu X.U. & Zhou G.P. 2001. Suport vector machines for predicting protein structural class. BMC Bioinformat. **2:** 3–7.

Cai Y.D. & Zhou G.P. 2000. Prediction of protein structural classes by neural network. Biochimie **82:** 783–785.

Cao Y., Liu S., Zhang L., Qin J., Wang J. & Tang K. 2006. Predicting of protein structural class with Rough Sets. BMC Bioinformat. **7:** 3–8.

Chan H.P. 1998. Computerized analysis of mammographic microcalcifications in morphological and texture feature spaces. Med. Phys. **25:** 2007–2019.

Chen C. 2006a. Predicting protein structural class with pseudo-amino acid composition and support vector machine fusion network. Anal. Biochem. **357:** 116–121.

Chen C.Y. 2006b. Using pseudo-amino acid composition and support vector machine to predict protein structural class. J. Theor. Biol. **243:** 444–448.

Chen K., Lukasz A. & Ruan J. 2008. Prediction of protein structural class using novel evolutionary collocation-based sequence representation. J. Comput. Chem. **29:** 1596–1604.

Chou K.C. 1995. A novel approach to predicting protein structural classes in a (20-1)-D amino acid composition space. Proteins **21:** 319–344.

Chou K.C. 1999. A key driving force in determination of protein structural classes. Biochem. Biophys. Res. Commun. **264:** 216–224.

Chou K.C. 2000. Prediction of protein structural classes and subcellular locations. Curr. Protein Pept. Sci. **1:** 171–208.

Chou K.C. 2005. Progress in protein structural class prediction and its impact to bioinformatics and proteomics. Curr. Protein Pept. Sci. **6:** 423–436.

Chou K.C. & Maggiora G.M., 1998. Domain structural class prediction. Protein Eng. **11:** 523–538.

Chou K.C. & Zhang C.T. 1994. Predicting protein folding types by distance functions that make allowances for amino acid interactions. J. Biol. Chem. **269:** 22014–22020.

Chou P.Y. & Fasman G.D. 1974. Prediction of protein conformation. Biochemistry **13:** 222–245.

Du Q.S. 2006. Amino acid principal component analysis (AAPCA) and its applications in protein structural class prediction. J. Biomol. Struct. Dyn. **23:** 635–640.

Feng K.Y. 2005. Boosting classifier for predicting protein domain structural class. Biochem. Biophys. Res. Commun. **334:** 213–217.

Garg A. & Gupta D. 2008. VirulentPred: a SVM based prediction method for virulent proteins in bacterial pathogens. BMC Bioinformat. **9:** 62.

Garnier J., Osguthorpe D. & Robson B. 1978. Analysis of the accuracy and implications of simple methods for predicting the secondary structure of globular proteins. J. Mol. Biol. **120:** 97–120.

Goldberg D.E. 1989. Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley Publishing Company, 432 pp.

Gromiha M.M. & Selvaraj S. 1998. Protein secondary structure prediction in different structural classes. Protein Eng. **11:** 249–251.

Ito M., Matsuo Y. & Nishikawa K. 1997. Prediction of protein secondary structure using the 3D-1D compatibility algorithm. Comput. Applic. Biosci. **13:** 415–423.

Jahandideh S., Abdolmalekia P., Jahandideh M. & Barzegari Asadabadi E. 2007a. Novel two-stage hybrid neural discriminant model for predicting proteins structural classes. Biophys. Chem. **128:** 87–93.

Jahandideh S., Abdolmalekia P., Jahandideh M. & Sadat Hayatshahia S.H. 2007b. Novel hybrid method for the evaluation of parameters contributing in determination of protein structural classes. J. Theor. Biol. **244:** 275–281.

King R.D. & Sternberg M.J.E. 1996. Protein secondary structure prediction based on position-specific scoring matrices. Protein Sci. **5:** 2298–2310.

Levitt M. & Chothia C. 1976. Structural patterns in globular proteins. Nature **261:** 552–557.

Mathew B.W. 1975. Comparison of the predicted and observed secondary structure of T4 phage lysozyme. Biochim. Biophys. Acta **405:** 442–451.

Metfessel B.A., Saurugger P.N., Connelly D.P. & Rich S.S. 1993. Cross-validation of protein structural class prediction using statistical clusturing and neural networks. Protein Sci. **2:** 1171–1182.

Murzin A.G., Brenner S.E., Hubbard T. & Chothia C. 1995. SCOP: a structural classification of proteins database for the investigation of sequences and structures. J. Mol. Biol. **247:** 536–540.

Nishikawa K., Kubota Y. & Ooi T. 1983a. Classification of proteins into groups based on amino acid composition and other characters. J. Biochem. **94:** 981–995.

Nishikawa K., Kubota Y. & Ooi T. 1983b. Classification of the proteins into groups based on amino acid composition and other characters grouping into four types. J. Biochem. **94:** 997–1007.

Nishikawa K. & Ooi T. 1982. Correlation of amino acid composition of a protein to its structural and biological characters. J. Biochem. **91:** 1821–1824.

Niu B. 2006. Predicting protein structural class with AdaBoost learner. Protein Peptide Lett. **13:** 489–492.

Orengo C.A., Michie A.D., Jones D.T., Swindells M.B. & Thornton J.M. 1997. CATH: a hierarchic classification of protein domain structures. Structure **5:** 1093–1108.

Qian W. 2005. Standardization for image characteristics in telemammography using genetic and nonlinear algorithms, Comput. Biol. Med. **35:** 183–196.

Rogen P. & Fain B. 2003. Automatic classification of protein structure by using Gauss integrals. Proc. Natl. Acad. Sci. USA **100:** 119–124.

Randall S.S. 2006. Knowledge discovery using a neural network simultaneous optimization algorithm on a real world classification problem. Eur. J. Oper. Res. **168:** 1009–1018.

Rost B. & Sander C. 1994. Combining evolutionary information and neural networks to predict protein secondary structure. Proteins **19:** 55–72.

Russell R.B. & Barton G.J. 1993. The limits of protein secondary structure prediction accuracy from multiple sequence alignment. J. Mol. Biol. **234:** 951–957.

Shenet H.B. 2005. Using supervised fuzzy clustering to predict protein structural classes. Biochem. Biophys. Res. Commun. **334:** 577–581.

Tantoso E. & Li K.B. 2008. AAIndexLoc: predicting subcellular localization of proteins based on a new representation of sequences using amino acid indices. Amino Acids **35:** 345–353.

Wang L. 2005. A hybrid genetic algorithm-neural network strategy for simulation optimization. Appl. Math. Comput. **170:** 1329–1343.

Xiao X.S. 2006. Using pseudo amino acid composition to predict protein structural classes: approached with complexity measure factor. J. Comput. Chem. **27:** 478–482.

Xiao X., Li W.Z. & Chou K.C. 2008a. Using grey dynamic modeling and pseudo amino acid components to predict protein structural classes. J. Comput. Chem. **29:** 2018–2024.

Xiao X., Wang P. & Chou K.C. 2008b. Predicting protein structural classes with pseudo amino acid composition: an approach using geometric moments of cellular automaton image. J. Theor. Biol. **254:** 691–696.

Yi T.M. & Lander E.S. 1993. Protein secondary structure prediction using nearest-neighbor methods. J. Mol. Biol. **232:** 1117–1129.

Zhou G.P. 1998. An intriguing controversy over protein structural class prediction. J. Protein Chem. **17:** 729–738.

Zhang, P. 2005. Neural vs. statistical classifier in conjunction with genetic algorithm based feature selection. Pattern Recogn. Lett. **26:** 909–919.

Received August 19, 2008
Accepted January 2, 2009