Amino Acids (2006) 30: 469–475
DOI 10.1007/s00726-005-0239-0

Amino Acids
Printed in Austria

Prediction of protein structural classes using support vector machines

X.-D. Sun and **R.-B. Huang**

College of Life Science and Biotechnology, Guangxi University, Nanning, Guangxi, China

Received June 1, 2005
Accepted July 12, 2005
Published online April 20, 2006; © Springer-Verlag 2006

Summary. The support vector machine, a machine-learning method, is used to predict the four structural classes, i.e. mainly α, mainly β, α–β and fss, from the topology-level of CATH protein structure database. For the binary classification, any two structural classes which do not share any secondary structure such as α and β elements could be classified with as high as 90% accuracy. The accuracy, however, will decrease to less than 70% if the structural classes to be classified contain structure elements in common. Our study also shows that the dimensions of feature space $20^2 = 400$ (for dipeptide) and $20^3 = 8\,000$ (for tripeptide) give nearly the same prediction accuracy. Among these 4 structural classes, multi-class classification gives an overall accuracy of about 52%, indicating that the multi-class classification technique in support of vector machines may still need to be further improved in future investigation.

Keywords: Support vector machines – CATH – Multi-class – Protein structural class prediction – Jackknifing

Introduction

It has been more than four decades since it had been explicitly elucidated that the amino acid sequence of proteins determined their three-dimensional (3D) structures (Sela et al., 1957). Proteins always manifest their functions through the three-dimensional structures. Chothia (1992) estimated that there were no more than 1 000 distinct protein structure entities (called folds) in nature, with which the whole range of diversity of functions in the kingdoms of life could be performed. It followed, through analysis of the protein interaction data, that the total number of protein interactions may be also quite limited, being just in the range of 10 000 types (Aloy and Russell, 2004). Even the total exons in the universal are calculated to be about 30 000 in size (Saxonon and Gilbert, 2003), based on the information collected from the databases of genomic sequences. It is therefore very important for researchers to acquire information on the protein structures because the interactions between different proteins or between proteins and their ligands (therefore they function through these interactions) are all determined by their 3D structures.

At present, there exists a severe "information asymmetry" between the researchers who are working on the sequencing of organism genomes and those in elucidating the 3D structure of biomolecules. On one hand, there are more and more genomes being sequenced, and on the other hand, protein structure information accumulation at Protein Data Bank (PDB) (Berman et al., 2002) is growing very slowly, compared with the sequence's one, since the structural determination in PDB is heavily relied on the experimental methods such as x-ray crystallography and NMR. These methods are expensive and time consuming, and many proteins such as trans-membrane proteins or some large proteins may not be amenable to these techniques. Because of this, protein structure prediction by homology modeling or computer simulating is therefore emerging as an alternative or complementary approach.

The core issue for computational prediction of protein structure is how to decipher the underlying mechanism that link protein sequence to its structure. If we take it for granted that protein structure entities (protein folds) are limited to around 1 000 as estimated above, then the sequences governing these structures should have some regular patterns to let one trace rather easily by using the methods of computational biology. Several computational methods have been applied in prediction of the structural classes of proteins from their amino acid sequences; these include amino acid composition (Chou, 1980, 1989; Nakashima et al., 1986; Klein and Delisi, 1986; Zhang and Chou, 1992; Dubchak et al., 1993; Metfessel et al., 1993; Rost and Sander, 1994; Chandonia and Karplus, 1995), Artificial Neural Network (ANN) (Rost and Sander, 1993), Hidden Markov Models (HMM) (Hubbard and Park, 1995) and Support Vector Machines (SVMs).

Among all protein structural class prediction methods developed so far, amino acid composition (AAC) and AAC with component-coupled effect are the most extensively studied ones (Chou and Zhang, 1995; Chou, 2000). Chou and colleagues pioneered an elegant work that incorporated the component-coupled effect with AAC, and coined this novel approach as component-coupled algorithm (Chou and Zhang, 1994, 1995; Chou, 1995). For the last decade, the component-coupled algorithm has been used to predict the protein structural classes and subcellular locations (Chou et al., 1998; Chou and Maggiora, 1998; Chou and Elrod, 1998, 1999a, b; Zhou, 1998). Unlike neural network methods, the component-coupled algorithm has very sound physical foundation for explaining the possible prediction success, which is very important to pinpoint the critical factors to be adjusted towards prediction improvement (Bahar et al., 1997; Chou et al., 1998; Chou, 1999; Zhou, 1998; Zhou and Assa-Munt, 2001). Recently, the functional domain composition approach was introduced that remarkably improved the prediction quality (Chou and Cai, 2004).

SVM is a machine learning technique that is based on the statistical learning theory developed by Vapnik (1995), and it is also considered to be the one of the best computer algorithms in this field (Yang, 2004). This is because SVMs are designed to maximize the margin to separate two classes, so that the trained model could generalize well on test data (Yang, 2004). Most other computer learning algorithms such as NN and HMM implement a classifier through minimization of error occurred in training, which will lead to poorer generalization on unseen data (Yang and Chou, 2004). SVMs have been widely used in the field of bioinformatics such as, among the others, in analysis of microarray gene data (Brown et al., 2000), in prediction of protein subcellular locations Cai et al., 2000), in protein secondary structure prediction (Hua and Sun, 2001), and in prediction of protein domain structure class (Cai et al., 2002, 2003b).

According to convention (Levitt and Chothia, 1976), a protein could be classified into one of four structural classes, based on its secondary structure components: all α, all β, $\alpha+\beta$ and α/β (Levitt and Chothia, 1976; Rechardson and Rechardson, 1989). Finding the protein secondary structures is the first step to build its 3D structure (Creighton, 1993). If the structural class of a protein is known, it can be used to considerably reduce the search space of structure prediction processes, since most of the structure alternatives could be eliminated and therefore the structure prediction task will be simplified and whole process will be accelerated (Isik et al., 2004).

This investigation involves the protein class prediction based on the database of hierarchic classification of protein domain structure (CATH) developed by Thornton group (Orengo et al., 1997). In CATH database, protein domains can be classified into four hierarchical levels. The first level is class (C), which consists mainly of four secondary structures known as mainly α, mainly β, mixed α–β and few secondary structure (Few SS). The second level is architecture (A), which describes the gross arrangement of secondary structures without taking into account the connectivity of the secondary structure units (α, β). The third level is topology (T-level), also called fold families. If proteins belong to the same T-level, they not only have the similar number and arrangement of secondary structures, but also the same connectivity linking their secondary structure elements. The fourth level is homologous superfamily (H-level), in which proteins will have highly similar structures and functional similarity. In this research, 820 folds taken from T-level are used as training data set from which four structural classes (mainly α, mainly β, mixed α–β and Few SS) are predicted using support vector machines, specially considering the impact of relationship between amino acid neighbors in the sequence, i.e., effects of the relative frequencies of dipeptide and tripeptides on the training results. Similar research has been carried out by Markowetz et al. (2003) and by Isik et al. (2004), but their data sets are different from the one used here, and so far current investigation is the first report on CATH database used for structural prediction by SVM. Furthermore, the data set of current research is far bigger than the both, i.e. 820 sequences against only $117+63$ and 268 sequences respectively. Leslie et al. (2002) used κ-spectrum of a protein sequence to create 20^{κ} dimensions of feature map and they used SCOP database as their training and testing data set.

Materials and methods

Support vector machines

Support vector machines (SVMs) are based on statistical learning theory which was first developed by Vapnik (1995) described in detail by Cristianini and Shawe-Taylor (2000). Elaborated treatment on SVMs can be found in Kecman (2001) and Scholkopf and Smola (2002). The principles underlying the SVMs have also been detailed by Chou and Cai (2002), Cai et al. (2002a, 2003b). Here, SVMs will only be treated very briefly. Let us consider a binary classification task with datapoints $x_i(i = 1, \cdots, m)$ having corresponding labels $y_i = \pm 1$. Each datapoint is represented in a dimensional input or attribute space. Let the classification function be: $f(x) = sign(w \cdot x + b)$. Here the vector w determines the orientation of a discriminant plane (hyperplane). The non-vector b determines the offset of the plane from the origin. The hyperplane should separate the data, so that $w \cdot x + b > 0$ for all x_i of one class, and $w \cdot x + b < 0$ for all the x_j of other class. If the data are separable in this way, there is probably more than one way to do it. Among the possible hyperplane, SVMs select the one where the distance of the hyperplane from the closest data points (the "margin") is as large as possible. How to choose the suitable hyperplane? Scholkopf and Smola (2002) describe an intuitive justification for the criterion: suppose the training data are good, in the sense that every possible test vector is within some radius r of a training vector. Then if the chosen hyperplane is at least r from any training vector it will correctly separate all the test data, r is allowed to be correspondingly large. The desired hyperplane (that maximizes the margin) is also bisector of the line between the closest points on the convex hulls of the data sets.

The main task for SVMs is in fact to find this hyperplane. To find it, label the training points by $y_i \in -1, 1$, with 1 being a positive example, -1 a negative training example.

$$y_i(w \cdot x + b) \geq 0, \text{ for all points}$$

Both w, b can be scaled without changing the hyperplane.

To remove this freedom, scale w, b so that

$$y_i(w \cdot x + b) \geq 1, \quad \forall i$$

For non-linear data input, SVMs will resort to kernel functions to correct the separating hyperplane (Scholkopf and Smola, 2002).

Program used in this research is Libsvm which applied simplified SVMs algorithm to both SMO (Platt, 1999) and SVMlight (Thorsten, 2002) and can be obtained from: http://www.csie.ntu.edu.tw/~cjlin/ libsvm/.

Data set

The training data set is taken from CATH database, which was built in January 28, 2004 and the Version number 2.5.1. In this version of CATH, there are 820 fold families at topology-level, representing some 48 000 domains. In each fold family the individual protein sequences should have empirical trial score (SSAP score, or Sequence Structure Alignment Program Score) 70 according to CATH's classification criteria (Taylor and Orengo, 1989), and there should be 60% of the larger protein matching the smaller protein.

Table 1. The training data set built by taking the protein sequences from each T-level fold family of 820 total topology families*

Classes	Number of fold families	Range of chain length
α	227	38–740
β	139	33–574
α–β	368	36–534
fss	86	16–119

* Data from CATH version 2.5.1: http://www.biochem.ucl.ac.uk/bsm/cath/releases.html

A protein sequence is manually picked up from each fold family to build a data set with 820 samples (Table 1).

Data embedding and SVMs training

Before carrying out SVMs data training, the protein sequence information taken from CATH database has to be represented in a way SVM program could process. These protein sequences are embedded into a more regularly defined feature space (Markowetz et al., 2003). A typical embedding is achieved by the relative frequencies of dipeptide and tripeptide amino acids. It is important to take dipeptide and tripeptide into account, since the way a protein folds does not only depend on single amino acids, but also on the relationships between neighbors in the sequence (Branden and Tooze, 1999). For 20 amino acids, if a dipeptide is considered each time, there would be $20^2 = 400$ possible dipeptides required to be treated. It is the same as for tripeptide situation, there would be $20^3 = 8\,000$ possible tripeptides, which will constitute an input space with 8 000 dimensions. Figure 1 shows the data embedding principle for running SVMs data training, taking CATH's small peptide sequence no. 4.10.180.10.1.1.1 as example, which has sequence composition of ASMWERVKSIIKSSLA.

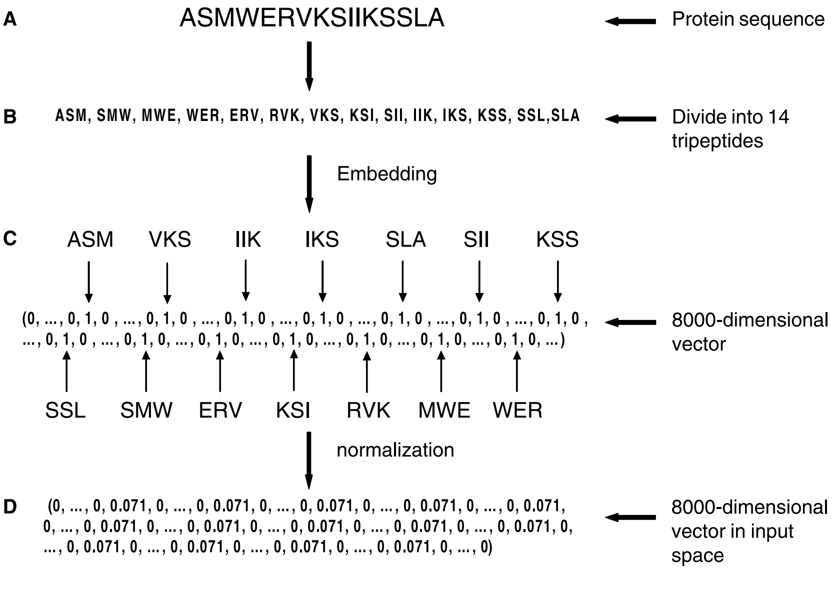

Fig. 1. Preparation of data input before SVM processing. **A** The sequence of short peptide taken from CATH database, and it belongs to few ss class. **B** The peptide is divided into tripeptides in all possible consecutive triplets. **C** The tripeptides are embedded into 8 000-dimensional vectors with occurrence of a tripeptide designed as 1, and absent position as zero. **D** The figures obtained in step C are normalized by dividing every figure with total number of tripeptides in this example sequence. For instance, this peptide was divided into 14 tripeptides, so every figure (position) is divided by 14, and $1/14 \doteq 0.071$

Cross validation

The evaluation of accuracy of SVMs prediction is necessary to estimate the performance of the method. The Leave One Out Cross-Validation (LOOCV) procedure, also known as Jackknife Test (Chou and Zhang, 1995), was carried out to test the accuracy of the prediction. In LOOCV, data are divided into two sets: the "training set" which is used to train the SVMs classifier, and the "test set" on which the trained classifier is tested. In this research, 820 polypeptides were separated into 820 subsets, i.e. each peptide represents a subset. LOOCV (Jackknife) test leaves one peptide out, and performs the training on 819 subsets (819 peptides), and testing was employed on the peptide that left out. This was repeated once for each of 820 protein chains, producing 820 test results. This is called full Jackknife Test and the accuracy is the percentage of subsets which is correctly predicted. This method uses all of the data for testing, but since the test data is not used for the corresponding training phase, the testing is hence unbiased (Chou and Zhang, 1995).

Multi-class prediction methods

SVMs is normally used in binary classification, but here we are using SVMs to do 4 classes (mainly α, mainly β, mixed $\alpha+\beta$ and FSS) classification. Therefore, the multi-class prediction method will be used in this case. There are many ways to apply SVMs to $n > 2$ classes, but all these different ways are still using two-class classification method as the basic building blocks. In this research, two multi-class classification schemes are employed: one is one-against-one classifier in which six binary classifiers are built: α vs. β, α vs. α–β, α vs. fss, β vs. α–β, β vs. fss and α–β vs. fss; another is one-against-others in which four binary classifiers are involved: α vs. others (including β, α–β and fss), β vs. others (α, α–β and fss), α–β vs. others (α, β, and fss) and fss vs. others (α, β and α–β) (Ding and Dubchak, 2001; Markowetz, 2002).

Results and discussion

One-against-one classification

As described in Materials and methods, one-against-one classification was carried out by the different combination of four structural classes. The prediction accuracy of each one-against-one classification is shown in Table 2.

From Table 2, it seems to be that the classifiers between any distinguished different structure classes will gain higher accuracy, comparing with the other classifiers, especially the "mixed structure" classes such as α–β vs. α, and α–β vs. β, in which they share some structural classes. So the classifiers α vs. β, α vs. fss, α–β vs. fss give over 80% accuracy, but SVMs give a slightly low accuracy when they classified any two classes which share certain similar structures. It can be seen from Table 2 that the classifiers α–β vs. α, and α–β vs. β give only about 70% accuracy, probably because there are structure similarities among the classes α and α–β, and among β and α–β. The only exception to this tendency is classifier β vs. fss which gives only 72–74% accuracy although belongs to the clearly-cut different structural classes. It also can be seen that there is little difference of prediction accuracy between dipeptide and tripeptide embeddings in our one-against-one classification.

Table 2. Prediction accuracy of one-against-one classification

Classifiers	Accuracy of dipeptide frequency (%)	Accuracy of tripeptide frequency (%)
α vs. β	81.69	80.27
α vs. α–β	71.89	68.01
α vs. fss	81.73	81.73
β vs. α–β	72.58	72.73
β vs. fss	72.77	74.11
α–β vs. fss	87.44	87.86
Average accuracy	78.02	77.45

One-against-others classification

There are four One-against-others classifications in our research and the prediction accuracy is represented in Table 3. It is quite apparent that there is little difference between the dipeptide and tripeptide frequencies, regarding to the prediction accuracy.

The classifier fss vs. others gives the highest prediction accuracy, being about 90%. The β vs. others also gives good accuracy, in the range of 80%–83%. It is interesting to see that the classifiers between any class that contains α structure and the "others" classes will produce considerably lower prediction accuracy. The classifier α vs. others, for example, only give about 75% accuracy, and the another classifier α–β vs. other, even produces poorer result, with only around 62% accuracy.

Table 3. The binary classification accuracies of class folds (one-against-other)

Classifier	Accuracies of dipeptide frequency (%)	Accuracies of tripeptide frequency (%)
α vs. other (including β, α–β and fss)	74.60	75.34
β vs. other (including α, α–β and fss)	80.83	83.03
α–β vs. other (including α, β and fss)	62.56	62.03
fss vs. other (including α, β and α–β)	89.74	90.23
Average accuracies	76.93	77.66

Multi-class classification

When SVMs are applied to real-world classification problems, the following optimization problem needs to be solved:

$$\min_{w,b,\xi} \frac{1}{2} W^T W + C \sum_{i=1}^{l} \xi_i$$

$$\text{subject to } y_i(W^T \phi(x_i) + b \geq 1 - \xi_i, \quad \xi_i \geq 0$$

Here, C is the penalty of parameter of error term. For binary classification, kernel functions are often needed to deal with nonlinear data relationship. For multi-class classification, use of kernels is sometime imperative in order to obtain reasonably good prediction accuracy. Among the kernels, the best choice could be radial basis function (RBF):

$$K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2), \quad \gamma > 0$$

In order to use the RBF kernel to classify our multi-class data, two parameters, C and γ, must be known before carrying out the training and testing. A grid search program, grid.py, by Chang and Lin (2001), is used to choose the suitable values of C and γ. Our results are shown in Figs. 2 and 3. In our data set, $C = 2^3$ and $\gamma = 2^3$ are used to train the data and conduct cross-validation testing. For the multi-classes of α, β, α–β and fss, the overall four-class prediction accuracy is 52.26% for dipeptide and 52.01% for tripepetide embedding respectively.

To test the power of a prediction method, it is necessary to conduct a cross-validation test. The cross-validation test mainly consists of three approaches, i.e., single independent dataset test, sub-sampling test, and a jackknife test (for a comprehensive review, see Chou and Zhang, 1995). In this study, only the jackknife test is carried out because it is the most rigorous and reliable one (Chou and Zhang, 1995), and has been used by more and more investigators (see, e.g., Zhou, 1998; Zhou and Assa-Munt, 2001; Zhou and Doctor, 2003).

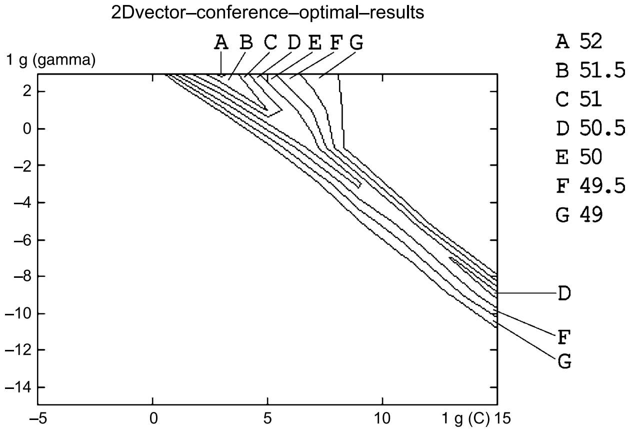

Fig. 2. Grid search for optimal values of C and γ for dipeptide embedding. The program used is grid.py and is a module inside the LibSVM package which can be freely downloaded from the internet. Any point, which represents a combination of C and γ, inside the areas A, B, C, D, E, F, G, and H, will give prediction accuracy from 52% to 49%, as shown in the graph

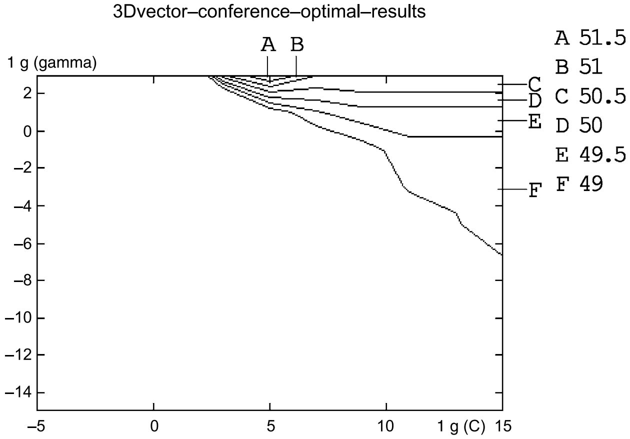

Fig. 3. Grid search for optimal values of C and γ for tripeptide embedding. See Fig. 2 for further details

In this study, CATH database was first trained and then tested using a newly developed machine-learning method SVMs. There are four classes of structures needed to be predicted from the training dataset. For 820 peptides at topology-level of CATH database, two multi-class prediction methods, i.e. one-against-one and one-against-others are used and their binary classifications are in the same range of about 77% accuracy. For dipeptide and tripeptide embedding, the prediction average accuracy is also nearly the same, being 78.02% and 77.45% respectively for one-against-one, and 76.93% and 77.66% respectively for one-ainst-others. To our surprise, the cross-validation of the multi-class classification gives much lower accuracy, overall accuracy being only 52.26% for dipeptide and 52.01% for tripeptide embedding respectively. The causes for the low accuracy need to be focused on in the next coming study.

In this investigation, special attention is paid to the dipeptide and tripeptide sampling. Since the amino acid sequence governs the 3D protein structure, the data embedding methods such as dipeptide and tripeptide sampling should give very good prediction accuracy because they have taken into account the amino acid neighboring effect. But in fact the average performance is not as good as the Component-Coupled Algorithm (Chou, 2000).

Although the conclusion like this may be drawn little too earlier without comparison of both methods on the same dataset, it seems still quite apparent that amino acid composition may play a very important role in determination of 3D structure (Du et al., 2003). Highly successful rates for the Component-Coupled Algorithm are reminiscent of the situation in nucleic acids. It has been long known that GC content in DNA is always constant for a particular species, or for strains in bacteria (Chargaff, 1951, 1979). In DNA composition, GC content differed in as few as 0.5% will probably mean a different bacterial strain. In another word, in DNA world, changes in GC content (expressed as %) seems to be more important than the changes such as mutations in the DNA sequence (Sueoka, 1961). If all facts above are combined together, that include the moderate prediction results from SVMs dipeptide and tripeptide sampling, high accuracy rate of the amino acid composition method with component-coupled effect, and the GC content sensitivity in the genomic DNA (Chargaf's GC rule), one may be able to conclude that just as base content is so critical in DNA composition, the amino acid composition probably play more important role than any other factors else. If this is true, then in this SVMs prediction study the dipeptide and tripeptide samplings give nearly the same accuracy would be easily explained. In fact, even using AAC with component-coupled effect, the contribution from the different samplings of di-, tri-, tetra-, penta- and hexa-peptides is moderate (maximum 20% gain) (Luo et al., 2002), implying that the amino acid composition may be more important factor than amino acid sequence in shaping protein structural class.

In the field of SVM's applications, binary classification may still be used from time to time, but the trends are quite apparent that main classification problems in the real world belong to multi-class one (Nguyen and Rajapaks, 2003). It is therefore very important for biologists to widely test multi-class classification on various databases. It is no doubt that in the near future, there are two active facets in the field of SVM's application in bioinformatics. One is to carry on the testing of multi-class classifications on different existing databases or newly created databases in order to accumulate more information on the strength of this machine-learning approach. Another is to continue on the development of new multi-class classification algorithms (Anguita et al., 2004). Since multi-class classification has capacity to solve the optimization problem in one step, it should become a hot spot in SVM's application research in the coming years.

Acknowledgements

The authors would like to thank Dr. Chih-Jen Lin for granting the permission to us to use the LiBSVM package. This work was supported by the project 2004CB719606 of the National Basic Research Program ("973") of China.

References

Aloy P, Russell R (2004) Ten thousand interactions for the molecular biologist. Nature Biotechnol 22: 1317–1321

Anfinsen CB (1973) Principles that govern folding of protein chain. Science 181: 223

Anguita D, Ridella S, Sterpi D (2004) A new method for multiclass support vector machines. In: Neural Networks, 2004. Proceedings 2004 IEEE International Joint Conference, pp 412–417

Berman HM, Battistuz T, Bhat TN, Bluhm WF, Bourne PE, Burkhardt K, et al. (2002) The protein data bank. Acta Cryst D 58: 899–907

Branden C, Tooze J (1999) Introduction to protein structure, 2nd ed. Garland Publishing, New York

Brown MPS, Grundy WN, Lin D, Cristianini N, Sugnet CW, Furey TS, et al. (2000) Knowledge-based analysis of microarray gene expression data by using support vector machine. Proc Natl Acad Sci USA 97: 262–267

Cai YD, Liu XJ, Xu XB, Chou KC (2000) Support vector machines for prediction of subcellular location. Mol Cell Biol Res Commun 4: 230–233

Cai YD, Liu XJ, Xu XB, Chou KC (2002a) Support vector machines for predicting the specificity of GalNAc-transferase. Peptides 23: 205–208

Cai YD, Liu XJ, Xu XB, Chou KC (2002b) Prediction of protein structure classes by support vector machines. Comput Chem 26: 293–296

Cai YD, Liu XJ, Li YX, Xu XB, Chou KC (2003a) Prediction of β-turns with learning machines. Peptides 24: 665–659

Cai YD, Liu XJ, Xu JB, Chou KC (2003b) Support vector machines for prediction of protein domain structural class. J Theor Biol 221: 115–120

Chandonia JM, Karplus M (1995) Neural networks for secondary structure and structural class prediction. Protein Sci 4: 275–285

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

Chargaff E (1951) Structure and function of nucleic acids as cell constituents. Fed Proc 10: 654–659

Chargaff E (1979) How genetics got a chemical education. Ann NY Acad Sci 325: 345–360

Chothia C (1992) One thousand families for the molecular biologist. Nature 357: 543–544

Chou JJ, Zhang CT (1993) A joint prediction of the folding types of 1490 human proteins from their genetic codons. J Theor Biol 161: 251–262

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

Chou KC (2000) Review: Prediction of protein structural classes and subcellular locations. Curr Protein Pept Sci 1: 171–208

Chou KC, Cai YD (2002) Using functional domain composition and support vector machines for prediction of protein subcellular location. J Biol Chem 277: 45765–45769

Chou KC, Cai YD (2004) Predicting protein structural class by functional domain composition. Biochem Biophys Res Comm 321: 1007–1009

Chou KC, Elrod DW (1998) Using discriminant function for prediction of subcellular location of prokaryotic proteins. Biochem Biophys Res Commun 252: 63–68

Chou KC, Elrod DW (1999a) Protein subcellular location prediction. Protein Eng 12: 107–118

Chou KC, Elrod DW (1999b) Prediction of membrane protein types and subcellular locations. Proteins 34: 137–153

Chou KC, Maggiora GM (1998) Domain structural class prediction. Protein Eng 11: 523–538

Chou KC, Liu W, Maggiora GM, Zhang CT (1998) Prediction and classification of domain structural classes. Proteins 31: 97–103

Chou KC, Zhang CT (1994) Predicting protein folding types by distance functions that make allowances for amino acid interactions. J Biol Chem 269: 22014–22020

Chou KC, Zhang CT (1995) Review: Prediction of protein structural classes. Crit Rev Biochem Mol Biol 30: 275–349

Chou PY (1980) Amino acid composition of four classes of proteins. In: Abstracts of Papers, Part I, Second Chemical Congress of the North American Continent, Las Vegas, Nevada

Chou PY (1989) Prediction of protein structural classes from amino acid composition. In: Fasman GD (ed) Prediction of protein structure and the principles of protein conformation. Plenum Press, New York, pp 549–586

Creighton T (1993) Proteins, structures and molecular properties, 2nd ed. Freeman and Company, New York

Cristianini N, Shawe-Taylor J (2000) An introduction to support vector machines. Cambridge University Press, Cambridge

Ding CHQ, Dubchak I (2001) Multi-class protein fold recognition using support vector machines and neural networks. Bioinformatics 17: 345–358

Dubchak I, Holbrook SR, Kim SH (1993) Predicting protein secondary structure content: a tandem neural network approach. Proteins 16: 79–91

Du QS, Wei DQ, Chou KC (2003) Correlations of amino acids in protein. Peptides 24: 1863–1869

Hua S, Sun S (2001) A novel method of protein secondary structure prediction with high segment overlap measure: support vector machine approach. J Mol Biol 302: 397–407

Hubbard TJP, Park J (1995) Fold recognition and ab initio structure predictions using Hidden Markov Models and b-strand pair potentials. Proteins 23: 398–402

Isik Z, Yanikoglu B, Sezerman U (2004) Protein structural class determination using support vector machines. In: Aykanat C, Dayar T, Korpeoglu I (eds) Lecture notes in computer science, vol 3280: Computer and information sciences. Springer, New York, pp 82–89

Kecman V (2001) Learning and soft computing. MIT Press, Cambridge

Klein P, Delisi C (1986) Prediction of protein structural class from amino acid sequence. Biopolymers 25: 1659–1672

Leslie C, Eskim E, Noble SW (2002) The spectrum kernel: a string kernel for SVM protein classification. In: Proc. Pacific Symposium on Biocomputing 7: 566–775

Levitt M, Chothia C (1976) Structural patterns in globular proteins. Nature 261: 552–558

Luo RY, Feng ZP, Liu JK (2002) Prediction of protein structural class by amino acid and polypeptide composition. Eur J Biochem 269: 4219–4225

Markowetz F, Edler L, Vingron M (2003) Support vector machines for protein fold class prediction. Biometr J 45: 377–389

Metfessel BA, Saurugger PN, Connelly DP, Rich ST (1993) Cross-validation of protein structural class prediction using statistical clustering and neural networks. Protein Sci 2: 1171–1182

Nakashima H, Nishikawa K, Ooi T (1986) The folding type of a protein is relevant to the amino acid composition. J Biochem 99: 152–162

Nguyen MN, Rajapakse JC (2003) Multi-class support vector machines for protein secondary structure prediction. Genome Informatics 14: 218–227

Orengo CA, Machie AD, Jones S, Jones DT, Swindells MB, Thornton SM (1997) CATH – a hierarchic classification of protein domain structures. Structure 5: 1093–1108

Platt JC (1999) Fast training of support vector machines using sequsntial minimal optimization. In: Scholkopf B, Verges CJC, Smola AJ (eds) Advances in kernel methods: support vector learning. MIT Press, Cambridge

Rechardson JS, Rechardson DC (1989) Principles and patterns of protein conformation. In: Fasman GD (ed) Prediction of protein structure and the principles of protein conformation. Plenum Press, New York, pp 1–98

Rost B, Sander C (1993) Prediction of protein secondary structure at better than 70% accuracy. J Mol Biol 232: 584–599

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

Saxonon S, Gilbert W (2003) The universal of exons revisited. Genetica 118: 267–278

Scholkopf B, Smola A (2002) Learning with kernels. MIT Press, Cambridge

Sela M, White FH Jr, Anfinsen CB (1957) Reductive cleavage of disulfide bridges in ribonuclease. Science 125: 691–692

Sueoka N (1961) Compositional correlations between deoxyribonucleic acid and protein. Cold Spring Hard Symp Quant Biol 26: 35–43

Taylor WR, Orengo CA (1989) Protein structure alignment. J Mol Biol 208: 1–22

Thorsten J (2002) Learning to classify text using support vector machines. Kluwer, Norwell

Vapnik V (1995) Statistical learning theory. Wiley, New York

Yang ZR (2004) Biological applications of support vector machines. Brief Bioinformatics 5: 328–338

Yang ZR, Chou KC (2004) Bio-support vector machines for computational proteomic. Bioinformatics 20: 735–741

Zhang CT, Chou KC (1992) An optimization approach to predicting protein structural class from amino acid composition. Protein Sci 1: 401–408

Zhou GP (1998) An intriguing controversy over protein structural class prediction. J Protein Chem 17: 729–738

Zhou GP, Assa-Munt N (2001) Some insights into protein structural class prediction. Proteins Struct Funct Genet 44: 57–59

Zhou GP, Doctor K (2003) Subcellular location prediction of apoptosis proteins. Proteins Struct Funct Genet 50: 44–48

Authors' address: Prof. Ri-Bo Huang, College of Life Science and Biotechnology, Guangxi University, 100 University Road, Nanning, Guangxi 530004, China,
Fax: +86 771 3238107, E-mail: priboh@gxu.edu.cn