Journal of Peptide Science
*J. Peptide Sci.* **8**: 297–301 (2002)
Published online in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/psc.401

# Support Vector Machines for the Classification and Prediction of $\beta$-Turn Types

**YU-DONG CAI,[a]* XIAO-JUN LIU,[b] XUE-BIAO XU[c] and KUO-CHEN CHOU[d]**

[a] Shanghai Research Centre of Biotechnology, Chinese Academy of Sciences, Shanghai, 200233, China
[b] Institute of Cell, Animal and Population Biology, University of Edinburgh, West Mains Road, Edinburgh EH9 3JT, UK
[c] Department of Computing Science, University of Wales, College of Cardiff, Queens Buildings, Newport Road, PO Box 916, Cardiff CF2 3XF, UK
[d] Computer-aided Drug Discovery, Upjohn Laboratories, Kalamazoo, Michigan 49001-4940, USA

*Received 1 March 2002*
*Accepted 5 April 2002*

Abstract: The support vector machines (SVMs) method is proposed because it can reflect the sequence-coupling effect for a tetrapeptide in not only a $\beta$-turn or non-$\beta$-turn, but also in different types of $\beta$-turn. The results of the model for 6022 tetrapeptides indicate that the rates of self-consistency for $\beta$-turn types I, I′, II, II′, VI and VIII and non-$\beta$-turns are 99.92%, 96.8%, 98.02%, 97.75%, 100%, 97.19% and 100%, respectively. Using these training data, the rate of correct prediction by the SVMs for a given protein: rubredoxin (54 residues, 51 tetrapeptides) which includes 12 $\beta$-turn type I tetrapeptides, 1 $\beta$-turn type II tetrapeptide and 38 non-$\beta$-turns reached 82.4%. The high quality of prediction of the SVMs implies that the formation of different $\beta$-turn types or non-$\beta$-turns is considerably correlated with the sequence of a tetrapeptide. The SVMs can save CPU time and avoid the overfitting problem compared with the neural network method. 

Keywords: $\beta$-turns; $\beta$-turn prediction; conformational prediction

## INTRODUCTION

In protein structure, turns are formed wherever polypeptide chains reverse their overall direction. They not only play an important role in information of the three dimensional structure but are also involved in functional activities such as molecular recognition. Of the different turns, $\beta$-turns, which are formed by four residues, are the most common in protein structure, comprising on average 25% of the residues [1,2]. Therefore, the prediction of $\beta$-turns is an important basis for the prediction of the secondary structure and function of proteins. Several methods based on different algorithms for predicting $\beta$-turns have been proposed in the past few years [3–9]. Chou and Blinn [10] took the residue-coupled effect into account and proposed a new residue-coupled model for the prediction of $\beta$-turns in proteins. According to Chou's research, the prediction quality is significantly improved in comparison with the prediction results reported previously. Cai *et al.* [11] have used self-organization neural networks to predict $\beta$-turn types in proteins by using Chou's data. But the neural network method uses too much CPU time and always gives overfitted results. In this paper, we applied Vapnik's support vector machine [12] for this problem, to try to save CPU time and to avoid the overfitting problem, and good results were obtained.

## SUPPORT VECTOR MACHINE

The support vector machine (SVM) is one type of learning machine based on statistical learning

*Correspondence to: Dr Yu-Dong Cai, Biomolecular Sciences Department, UMIST, P.O. Box 88, Manchester, M60 1QD, UK; e-mail: y.cai@umist.ac.uk

theory. The basic idea of applying SVM to pattern classification can be stated briefly as follows. First, the input vectors are mapped into one feature space (possible with a higher dimension), either linearly or non-linearly, which is relevant to the selection of the kernel function. Then, within the feature space from the first step, an optimized linear division is found i.e. construct a hyperplane which separates two classes (this can be extended to multi-class). SVM training always seeks a global optimized solution and avoids overfitting, so it has the ability to deal with a large number of features. A complete description of the theory of SVMs for pattern recognition is in Vapnik's book [13].

SVMs have been used in a range of problems including drug design [14], image recognition and text classification [15].

In this paper, Vapnik's support vector machine [12] was applied for predicting $\beta$-turn types in proteins. The SVMlight was downloaded, which is an implementation (in C Language) of SVM for the problem of pattern recognition. The optimization algorithm used in SVMlight has been described by Joachims [16,17]. The code has been used in text classification and image recognition [15].

Suppose we are given a set of samples, i.e. a series of input vectors

$$X_i \in R^d (i = 1, \ldots, N)$$

with corresponding labels $y_i \in \{+1, -1\} (i = 1, \ldots, N)$.

Where $-1$ and $+1$ are used to stand respectively for the two classes. The goal here is to construct a one binary classifier or to derive a one decision function from the available samples, which has a small probability of misclassifying a future sample. Both the basic linear separable case and the most useful linear non-separable case for the most real life problems are considered here.

## THE LINEAR SEPARABLE CASE

In this case, a separating hyperplane exists whose function is $\vec{W} \bullet \vec{X} + b = 0$, which implies

$$y_i(\vec{W} \bullet \vec{x}_i + b) \geq 1, i = 1, \ldots, N$$

By minimizing $\frac{1}{2}\left\|\vec{W}\right\|^2$ subject to this constraint, the SVM approach tries to find a unique separating hyperplane. Here $\left\|\vec{w}\right\|^2$ is the Euclidean norm of $\vec{w}$, which maximizes the distance between the hyperplane (optimal separating hyperplane or OSH [18]) and the nearest data points of each class. The classifier is called the largest margin classifier. By introducing Lagrange multipliers $\alpha_i$, using the Karush-Kuhn-Tucker (KKT) conditions and the Wolfe dual theorem of optimization theory, the SVM training procedure amounts to solving the following convex QP problem

$$Max : \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j \cdot y_i y_j \cdot \vec{X}_i \bullet \vec{X}_j$$

subject to the following two conditions

$$\alpha_i \geq 0$$

$$\sum_{i=1}^{N} \alpha_i y_i = 0, i = 1, \ldots, N.$$

The solution is a unique globally optimized result that can be shown to have the following expansion

$$\vec{W} = \sum_{i=1}^{N} y_i \alpha_i \cdot \vec{x}_i.$$

Only if the corresponding $\alpha_i > 0$, then these $\vec{x}_i$ are called support vectors

When a SVM is trained, the decision function can be written as

$$f(\vec{x}) = \mathrm{sgn}\left(\sum_{i=1}^{N} y_i \alpha_i \cdot \vec{x} \bullet \vec{x}_i + b\right)$$

Where sgn() in the above formula is the given sign function.

## THE LINEAR NON-SEPARABLE CASE

Two important techniques needed for this case are given respectively as:

(i) 'Soft margin' technique.

In order to allow for training errors, Cortes and Vapnik [18] introduced slack variables

$$\xi_i > 0, i = 1, \ldots, N$$

and the relaxed separation constraint is given as

$$y_i(\overline{w} \bullet \vec{x}_i + b) \geq 1 - \xi_i, (i = 1, \ldots, N)$$

and the OSH can be found by minimizing

$$\frac{1}{2}|\overline{w}|^2 + C\sum_{i=1}^{N}\xi_i$$

instead of $\frac{1}{2}|\overline{w}|^2$ for the above two constraints in the previous section.
where $C$ is a regularization parameter used to decide a trade-off between the training error and the margin.

(ii) 'Kernel substitution' technique.

SVM performs a non-linear mapping of the input vector $\overline{x}$ from the input space $R^d$ into a higher dimensional Hilbert space, where the mapping is determined by the kernel function. Then as in case (i), it finds the OSH in the space $H$ corresponding to a non-linear boundary in the input space.

Two typical kernel functions are listed below

$$K(\vec{x}_i, \vec{x}_j) = (\vec{x}_i \bullet \vec{x}_j + 1)^d$$
$$K(\vec{x}_i, \vec{x}_j) = \exp(-r|\vec{x}_i - \vec{x}_j|^2).$$

Where the first one is called the polynomial kernel function of degree $d$ which will eventually revert to the linear function when $d = 1$, the latter is called the RBF (radial basic function) kernel.

Finally, for the selected kernel function, the learning task amounts to solving the following QP problem,

$$Max: \sum_{i=1}^{N}\alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j \cdot y_i y_j \cdot K(\vec{X}_i \bullet \vec{X}_j)$$

subject to

$$0 \le a_i \le C$$
$$\sum_{i=1}^{N}\alpha_i y_i = 0, i = 1, \ldots, N$$

and the form of the decision function is

$$f(\vec{x}) = \mathrm{sgn}\left(\sum_{i=1}^{N} y_i\alpha_i \cdot K(\vec{x}, \vec{x}_i) + b\right).$$

For a given data set, only the kernel function and the regularity parameter $C$ must be selected to specify one SVM.

## THE TRAINING AND PREDICTION OF $\beta$-TURN TYPES

Following the same procedures and rationale [10], the $\beta$-turn types classified by Hutchinson and Thornton [9] were clustered into seven categories, i.e. type I $\beta$-turn, type I′ $\beta$-turn, type II $\beta$-turn, type II′ $\beta$-turn, type VI $\beta$-turn, type VIII $\beta$-turn and non-$\beta$-turn. S1 was used to represent the dataset consisting of type I $\beta$-turn tetrapeptides, S1′ type I′ $\beta$-turn tetrapeptides, S2 type II′ $\beta$-turn tetrapeptides, S2′ type II′ $\beta$-turn tetrapeptides, S6 type VI $\beta$-turn tetrapeptides, S8 type VIII $\beta$-turn tetrapeptides and S- non $\beta$-turn tetrapeptides.

Since $\beta$-turn structure in a protein is a tetrapeptide that involves four consecutive residues i, i + 1, i + 2, and i + 3, its sequence can be generally expressed by Ri Ri + 1Ri + 2Ri + 3, where Ri represents the amino acid at the protein sequence position i (or subsite 1 of the tetrapeptide), Ri + 1 represents the amino acid at the protein sequence position $i + 1$ (or subsite 2 of the tetrapeptide), and so forth. For the current research, a tetrapeptide can be classified into one of the seven categories, as denoted by seven different sets: S1, S1′, S2, S2′, S6, S8 and S- as defined by Chou and Blinn [10].

Given a tetrapeptide, its assignment to which category sets(S1, S1′, S2, S2′, S6, S8 and S-) can be formulated by a 4-D (dimension) vector.

In this research, 20 bases of tetrapeptides are coded as 20-D vectors composed of only 0 and 1 (A = 100000...000, C = 010000...000, .....Y = 000000...001), which are taken as the input of SVMs.

The computations were carried out on a Silicon Graphics IRIS Indigo work station (Elan 4000).

There are 6028 $\beta$-turn tetrapeptides of $\beta$-turn types I(1227), I′(125), II(405), II′(89), VI(55), VIII(320) and non-$\beta$-turns(3807) in the training database. In this research, for the SVMs, the width of the Gaussian RBFs [13] is selected as that which minimized an estimate of the VC-dimension [13]. The parameter $C$ that controls the error-margin tradeoff is set at 100. After being trained, the hyperplane output by the SVMs was obtained. This indicates that the trained model, i.e. hyperplane output which includes the important information, has the function of identifying the $\beta$-turns.

In this research, as an example, $\beta$-turns and their types were predict from the entire primary

sequence of rubredoxin (54 residues, 51 tetrapeptides) including 12 $\beta$-turn types I tetrapeptides, 1 $\beta$-turn types II tetrapeptides and 38 non-$\beta$-turns. As a result, the prediction rate was quite high.

Because the training dataset and the extensive details for each classification (number of support vectors, the list of support vectors) are quite long, they are not detailed in this paper, but they are available upon request.

## RESULTS AND DISCUSSION

In this research, an examination for self-consistency of the SVMs method was tested. The rates of correct prediction for the seven classes reached 99.92%(types I), 96.8% (types I′), 98.02%(type II), 97.75% (type II′), 100%(type VI), 97.19%(type VIII) and 100% (non-$\beta$-turns). This indicates that after being trained, the hyperplanes output of the SVMs grasped the complicated relationship between the tetrapeptides and $\beta$-turns, and it can predict the unknown tetrapeptides. The SVMs only used about 20 min of CPU time for the whole training procedure.

In order to make a further test on the established model, we experimented on rubredoxin (54 residues) whose primary sequence is given by

MKKYTCTVCGYIYDPEDGDPDDGVNPGTD

FKDIPDDWVCPLCGVGKDEFEEVEE

Along with the sequence, $54 - 4 + 1 = 51$ tetrapeptides were automatically extracted in succession, and predicted by the SVMs model. The correct prediction rate of the $\beta$-turns in rubredoxin and their types achieved $49/51 = 96.1\%$. Compared with the results of the previous methods, such as $42/81 = 82.4\%$ [10] and $46/51 = 90.2\%$ [11], the results reported indicated a significant improvement.

In this research, on comparison with the neural network method, the following points can be made:

(1) For the self-consistency test, SVMs method only used about 20 min of CPU time, while the neural network method used about 25 h. This indicates that SVMs can save much CPU time.
(2) The rate of correct prediction (self-consistency) of the neural network method reached 100% [11] for each class, which is better than the results of SVMs (see above), but the rate for prediction of the neural network reached 90.2% [11], which is worse than the SVMs (rate = 96.1%). This indicates that the SVMs avoids the overfitting problem.

## CONCLUSION

The results using the SVMs research indicate that the formation of different $\beta$-turn types or non-$\beta$-turns is considerably correlated with the sequence of a tetrapeptide, fully consistent with the earlier report using a different approach [10,11] and we gained a significant improvement by using SVMs.

## REFERENCES

1. Kabsch W, Sander C. Dictionary of protein secondary structure: Pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers* 1983; **22**: 2577–2637.
2. Rose GD, Gierasch LM, Smith JA. Turns in peptides and proteins. *Adv. Protein Chem.* 1985; **37**: 1–109.
3. Lewis PN, Momany FA, Scheraga HA. Chain reversals in proteins. *Biochem. Biophys. Acta* 1973; **303**: 211–229.
4. Garnier J, Osguthorpe DJ, Robson B. Analysis of the accuracy and implications of simple methods for predicting the secondary structure of globular proteins. *J. Mol. Biol.* 1978; **120**: 97–120.
5. Chou PY, Fasman GD. Conformational parameters for amino acids in helical, $\beta$-sheet and random coil regions calculated from proteins. *Biochemistry* 1974; **13**: 211–223.
6. Cohen FE, Abarbanel RM, Kuntz ID, Fletterick RJ. The prediction in proteins using a pattern-matching approach. *Biochemistry* 1983; **25**: 266–275.
7. Wilmot CM, Thornton JM. Analysis and prediction of the different types of $\beta$-turn in proteins. *J. Mol. Biol.* 1988; **203**: 221–232.
8. McGregor MJ, Flores TP, Sternberg MJE. Prediction of $\beta$-turns in proteins using neural networks. *Protein Engineering* 1989; **2**: 521–526.
9. Hutchinson EG, Thornton JM. A revised set of potentials for $\beta$-turn formation in proteins. *Protein Sci.* 1994; **3**: 2207–2216.
10. Chou KC, Blinn JR. Classification and prediction of $\beta$-turn types. *J. Protein Chem.* 1997; **16**(6): 575–595.
11. Cai Yu-Dong, Li Yi-Xue, Chou Kuo-Chen. Classification and prediction of $\beta$-turn types by neural network. *Adv. in Engineer. Software* 1999; **30**: 347–352.

 

12. Vapnik V. *The Nature of Statistical Learning Theory*. Springer: Berlin, 1995.
13. Vapnik V. *Statistical Learning Theory*. Wiley-Interscience: New York, 1998.
14. Burbidge R, Trotter M, Holden S, Buxton B. Drug design by machine learning: support vector machine for pharmaceutical data analysis. *Proceedings of the AISB'00 Symposium on Artificial Intelligence in Bioinformatics*. Birmingham, 2000; 1–4.
15. Joachims T. Text categorization with support vector machines: learning with many relevant features. *Proceedings of the European Conference on Machine Learning*. Springer: Berlin, 1998.
16. Joachims T. 11 in: Making large-scale SVM Learning Practical. *Advances in Kernel Methods — Support Vector Learning*, Schölkopf B, Burges C, Smola A (eds). MIT Press. 1999.
17. Joachims T. Transductive inference for text classification using support vector machines. *International Conference on Machine Learning (ICML)*. Bled: Slovenia, 1999.
18. Cortes C, Vapnik V. Support vector networks. *Machine Learning* **20**: 1995; 273–293.

 