

SUPPORT VECTOR MACHINES FOR PREDICTION AND ANALYSIS OF BETA AND GAMMA-TURNS IN PROTEINS

THO HOAN PHAM*, KENJI SATOU*,[†] and TU BAO HO*,[†]

**Japan Advanced Institute of Science and Technology, 1-1 Asahidai, Tatsunokuchi, Ishikawa 923-1292, Japan*

[†]*Institute for Bioinformatics Research and Development (BIRD), Japan Science and Technology Agency (JST)
{h-pham,ken,bao}@jaist.ac.jp*

Received 10 June 2004

1st Revision 22 June 2004

2nd Revision 6 August 2004

Accepted 16 August 2004

Tight turns have long been recognized as one of the three important features of proteins, together with α -helix and β -sheet. Tight turns play an important role in globular proteins from both the structural and functional points of view. More than 90% tight turns are β -turns and most of the rest are γ -turns. Analysis and prediction of β -turns and γ -turns is very useful for design of new molecules such as drugs, pesticides, and antigens. In this paper we investigated two aspects of applying support vector machine (SVM), a promising machine learning method for bioinformatics, to prediction and analysis of β -turns and γ -turns. First, we developed two SVM-based methods, called BTSVM and GTSVM, which predict β -turns and γ -turns in a protein from its sequence. When compared with other methods, BTSVM has a superior performance and GTSVM is competitive. Second, we used SVMs with a linear kernel to estimate the support of amino acids for the formation of β -turns and γ -turns depending on their position in a protein. Our analysis results are more comprehensive and easier to use than the previous results in designing turns in proteins.

Keywords: β -turns; γ -turns; protein secondary structure; position-specific scoring matrices; support vector machine.

1. Introduction

Tight turn²⁵ play an important role in protein folding and stability. Tight turns are classified as σ -turns, γ -turns, β -turns, α -turns, and π -turns. About 90% of turns in proteins constitute β -turns and most of the remaining turns are γ -turns.¹² A β -turn is a four-residue reversal in a protein chain that is not in an α -helix, and the distance between $C_{\alpha}(i)$ and $C_{\alpha}(i+3)$ is less than 7 \AA .^{22,23} β -turns may or may not be accompanied by the $NH(i+3) - CO(i)$ hydrogen bond connecting the main-chain atoms. In contrast, a γ -turn consists of three consecutive residues at positions $i, i+1, i+2$, defined by the existence of a hydrogen

bond between the $CO(i)$ group and $NH(i + 2)$ group. β -turns and γ -turns provide very useful information for defining template structures for the design of new molecules such as drugs, pesticides, and antigens.⁵

There have been some attempts to predict and analyze β -turns and γ -turns. They can be divided into two categories: statistical and machine learning methods. The majority of statistical methods empirically employed the knowledge of amino acid preferences at individual positions in β -turns and γ -turns.^{6,7,27,28} Machine learning-based methods have been recently developed for prediction of β -turns and γ -turns. They include BTPRED,²⁴ BetaTPred2¹⁹ and GammaPred,¹⁸ which all use the neural network technique and multiple sequence alignment. These methods significantly outperformed statistical approaches. However, the prediction and analysis results are still restricted due to the complexity of the problem and the unbalanced nature of the data (especially γ -turn data).

In this paper, we introduce another machine learning approach, using support vector machine (SVM) for both prediction and analysis of β -turns and γ -turns. SVM is based on statistical learning theory and was developed by Vapnik.²⁶ In practice, SVM has a good performance and is easier to implement and train than neural networks. SVM has also been successfully applied to some problems in bioinformatics, such as secondary structure prediction,²⁰ microarray data analysis,¹³ protein-protein interactions,²¹ etc.

Two aspects of applying SVM to prediction and analysis of β -turns and γ -turns have been investigated in this research. First, we developed two SVM-based methods, BTSVM and GTSVM, that predict β -turns and γ -turns in a protein from its sequence. The prediction can be done with single sequence or multiple sequence alignment. The prediction results, on the dataset of 426 non-homologous protein chains by seven-fold cross-validation with BTSVM and on the dataset of 320 non-homologous protein chains by five-fold cross-validation with GTSVM, showed that our methods performed very well when compared to the other methods. Furthermore, the prediction results of our methods were improved when combined with additional secondary structure information, which is in turn predicted by another high accuracy secondary structure prediction method PSIPRED.¹⁵ Moreover, our methods performed the prediction at the turn level, which makes the prediction results more comprehensive and easier to interpret.

Second, we analyzed β -turns/ γ -turns by proposing the concept of “the support of an amino acid position for the formation of β -turns/ γ -turns under a linear SVM classification model” (we will refer to it as the support of an amino acid position in the rest of this paper), which implies both the contribution and prevention of that amino acid position for the formation of β -turns/ γ -turns. This information can be easily extracted from the “multivariable” classification model of a trained linear SVM. This model is more general than previously proposed models for prediction and analysis of β -turns and γ -turns such as Site-Independent model,⁸ 1–4 and 2–3 Residue-Correlation model,²⁸ and Sequence-Couple model.⁶ Our analysis

results, based on the supports of amino acid positions, are more comprehensive and easier to use than the previous ones.

Our methods for predicting β -turns and γ -turns with high accuracy and our easily understandable analysis results will be helpful for the researchers working in the fields of fold recognition and design of new molecules. We provide the web service for predicting β -turns and γ -turns at <http://genic.jaist.ac.jp/proteins>. *Related work:* There has been a work applying support vector machines for predicting different types of β -turns done by Cai *et al.*¹⁶ However, they used a single sequence as an input of their system, which would be much worse than multiple sequence alignment.^{17,19} Moreover, applying SVM to each of seven types of β -turns might have not good performance due to the very unbalanced data problem like γ -turns in our experiments. In this paper, we developed SVM-based methods using multiple sequence alignment for predicting general β -turns and γ -turns. Furthermore, we proved that SVMs can be useful for discovering of the support of amino acids for the formation of β -turns and γ -turns depending on their position in the protein sequence.

2. Materials and Methods

2.1. Datasets

We used the two datasets described in the work of Guruprasad and Rajkumar.¹² The first one (dataset *B*) consists of 426 non-homologous protein chains, while the second one (dataset *G*) consists of 320 non-homologous protein chains. These datasets have been used by Kaur and Raghava for assessing the performance of β -turn and γ -turn prediction methods.¹⁷⁻¹⁹ In each dataset, there are no two protein chains having more than 25% sequence identity. The structure of these proteins is determined by X-ray crystallography at resolutions higher than 2.0 Å. Each chain in the datasets contains at least one β -turn or γ -turn. The program PROMOTIF¹⁴ has been used to assign β -turns and γ -turns in these proteins. The datasets are available at <http://genic.jaist.ac.jp/proteins>.

2.2. Vector representations of a protein sequence

There are two basic ways to represent a protein sequence as a vector:

- (1) *Single sequence:* Each residue in the protein is represented by a 20-dimensional vector of 0 and 1 coding for the corresponding amino acid at this residue. This binary representation can be extended by taking into account the general substitute abilities (scores) of amino acids, i.e., BLOSUM62. Therefore, each residue is represented by a 20-dimensional vector of the substitute scores of 20 amino acids for this residue.
- (2) *Multiple sequence alignment:* A protein sequence is firstly aligned with a non-redundant (NR) database (e.g., the version used in our work contains 1, 109, 366 sequences) to find the family of sequences to which that protein belongs. The

alignment can be expressed in a scoring matrix of probability estimates or scores.² Two kinds of such matrices are considered in our work: position-specific frequency matrices (PSFMs) and position-specific scoring matrices (PSSMs). A PSFM is a table that lists the frequencies of each amino acid in the alignment, while a PSSM gives the log-odds score for finding a particular matching amino acid in a target sequence (see the work of Gribskov *et al.*¹¹ and Altschul *et al.*^{1,2} for more details). The stand-alone version 2.2.6 of PSI-BLAST (<ftp://ncbi.nlm.nih.gov/blast/executables/>) has been used to generate PSFMs and PSSMs in this work with **E-value** threshold of 0.001, three iterations and other parameters set to the respective default values.

In these two ways, each protein sequence is represented as a bi-dimensional vector $L \times 20$, where L is the length of the sequence. In this work, all elements of bi-dimensional vectors are scaled into the interval $[-1, 1]$ by a simple linear transformation function. After having vector representations of proteins, we use a sliding window with a fixed length w along each protein to extract the dataset of vectors and input them into the machine learning system (i.e., support vector machine).

2.3. Binary support vector machine

Support vector machine (SVM) is a learning technique based on statistical learning theory. The basic idea of applying SVM to binary pattern classification can be stated briefly as follows. First, map the input vectors into a feature space (often with a higher dimension), either linearly or non-linearly, which is relevant to the selection of the kernel function. Second, seek an optimized linear division within the feature space from the first step, i.e., construct a hyperplane which separates two classes.

The implementation of SVM is as follows. Suppose that $(x_i, y_i), i = 1, \dots, l$ be a training dataset, where x_i is a vector and $y_i = 1$ or -1 is a class attribute. SVM training solves the following problem:

$$\begin{aligned} \min_{w,b,\xi} \quad & \frac{1}{2}w^T w + C \sum_{i=1}^l \xi_i \\ & y_i(w^T \phi(x_i) + b) \geq 1 - \xi_i \\ & \xi_i \geq 0, \quad i = 1, \dots, l. \end{aligned}$$

Its dual is a quadratic optimization problem:

$$\begin{aligned} \min_{\alpha} \quad & \frac{1}{2}\alpha^T Q\alpha - e^T \alpha \\ & 0 \leq \alpha_i \leq C \\ & y^T \alpha = 0, \end{aligned}$$

where e is the vector of all ones; $C > 0$ is a error penalty parameter, $y = \{y_i\}_{i=1,\dots,l}$, $Q_{ij} = y_i y_j K(x_i, x_j)$, $K(x_i, x_j) = \phi(x_i)^T \phi(x_j)$ is a kernel function; and $\phi(x_i)$ maps x_i into a higher (maybe infinite) dimensional space. So $K(x_i, x_j)$ is a symmetric

positive definite function that reflects the similarity between the sample x_i and the sample x_j . In our research, we employed a linear function $K(x_i, x_j) = x_i \cdot x_j$ and radial basis function (RBF) $K(x_i, x_j) = \exp(-\gamma(x_i - x_j)^2)$ as the kernel functions. The SVM classification function, after trained, has the following form:

$$f(x) = \sum_i \alpha_i y_i K(x, x_i) + b \quad (1)$$

where $\alpha = \{\alpha_i\}_{i=1, \dots, l}$ is the solution of the above dual problem and b is in the solution of the prime problem. Based on the Karush–Kuhn–Tucker theory, the solution of the prime problem and that of its dual satisfy the following equation:

$$\alpha_i \{y_i (w^T \phi(x_i) + b) - 1 + \xi_i\} = 0.$$

Therefore, if there is an i such that $\alpha_i \neq 0$, then $y_i (w^T \phi(x_i) + b) - 1 + \xi_i = 0$. In this case, x_i is called a “support vector”.

SVM has a solid theoretical background, a good performance in practice, and a guaranteed global optimum. It can also handle a large dataset and is easier to implement and train than a neural network. A more detailed description of SVM can be found in the work of Vapnik²⁶ and Cristianini.⁹

2.4. Assigning positive and negative examples

To predict and analyze β -turns and γ -turns, we use a sliding window along the protein representation to get examples in a vector format. How to define positive and negative examples is an important issue. There are two options (Fig. 1):

- (1) *Assigning positive and negative examples at a residue level:* A window will be considered as a positive or negative example if its central residue falls in a turn area or not [Fig. 1(a)]. That is, in the training phase, a window with the central residue falling in a turn area will be considered as a positive example, otherwise negative. In the testing phase, the prediction result of a window will conversely be assigned only for one central residue. In this way, the results of prediction may be invalid and unclear when the number of turn-predicted consecutive residues do not fit into a β -turn/ γ -turn. For example, it is unrealistic to have three consecutive residues predicted as “ntn” or “tnt” (t for turn and n for non-turn). And it will be ambiguous to interpret the prediction result when more than five consecutive residues are predicted as β -turns/ γ -turns, like “ttttttttt”. How many β -turns or γ -turns are in this example? And where is the beginning of these turns?
- (2) *Assigning positive and negative examples at a turn level:* A window will be considered as a positive example if its four (or three with γ -turn) central residues form a β -turn/ γ -turn, otherwise negative [Fig. 1(b)]. In the training phase, a sliding window with four (or three with γ -turn) central residues forming a β -turn/ γ -turn will be considered as a positive example, otherwise negative. In the testing phase, if a window is classified as a positive example, it means that its four (or three with γ -turn) central residues are predicted

Sequence with 3 β -turns nnnnnnTtttnnnTtTtttnnnn	
nnnnnnTtt → 0	nnnnnnTttt → 0
nnnnnTttt → 0	nnnnnTtttn → 0
nnnnTtttn → 1	nnnnTtttnn → 0
nnnTtttnn → 1	nnnTtttnnn → 1
nnTtttnnn → 1	nnTtttnnnT → 0
nTtttnnnT → 1	nTtttnnnTt → 0
TtttnnnTt → 0	TtttnnnTtT → 0
tttnnnTtT → 0	tttnnnTtTt → 0
ttnnnTtTt → 0	ttnnnTtTtt → 0
tnnnTtTtt → 1	tnnnTtTttt → 0
nnnTtTttt → 1	nnnTtTtttn → 1
nnTtTtttn → 1	nnTtTtttnn → 0
nTtTtttnn → 1	nTtTtttnnn → 1
TtTtttnnn → 1	TtTtttnnnn → 0
tTtttnnnn → 1	tTtttnnnnn → 0
Ttttnnnnn → 0	
<div style="display: flex; justify-content: space-around;"> <div style="text-align: center;"> <p><i>a) At residue level</i></p> <p>(Sliding window size=9)</p> </div> <div style="text-align: center;"> <p><i>b) At turn level</i></p> <p>(Sliding window size=10)</p> </div> </div>	

Fig. 1. Assigning positive and negative windows at the residue level and turn level.

Table 1. The number of positive and negative examples at the residue level and the turn level.

Dataset	Level	#positive examples	#negative examples
Dataset <i>B</i> (426 proteins)	residue	23555	72358
	turn	7185	88728
Dataset <i>G</i> (320 proteins)	residue	2669	79566
	turn	904	81331

as a 4-residue- β -turn (3-residue- γ -turn). We used the signs “Tttt” for a 4-residue- β -turn and “Ttt” for a 3-residue- γ -turn, where T means the beginning of a turn and t means not-beginning of the turn. By using this approach in our work, we overcome the problems explained above.

We reported the number of positive and negative examples at the residue level and turn level in the datasets of β -turns (*B*) and γ -turns (*G*) in Table 1.

2.5. SVM method for discovering the support of attributes

Ranking informative (discriminant) attributes is of fundamental and practical interest in data mining and knowledge discovery. SVM has been successfully applied to this task.^{4,13} When SVM uses a linear kernel, it finds an optimal hyperplane that separates the positive from the negative class in the original space (not mapping into a higher dimensional space). This optimal hyperplane has then the following form (replacing $K(x, y) = x \cdot y$ in Eq. (1)):

$$f(X = (f_1, f_2, \dots, f_m)) = \sum_{i=1}^m w_i f_i + b. \quad (2)$$

We can change the signs of the weights $w_i, i = 1, \dots, m$, and b in the above function such that if $f(X) > 0$ then X would be classified as a positive example and otherwise negative. It can be clearly seen that if w_i is positive, the attribute i would support the positive class; otherwise this attribute would support the negative class (or prevent the positive class); and the larger the absolute value of w_i , the stronger the attribute i supports (or prevents). From this remark, we define the weight w_i as the *support of the attribute i* .

2.6. Performance measures

We use four criteria described in the work of Shepherd *et al.*²⁴: (1) Q_{total} (prediction accuracy), the percentage of correctly predicted residues, (2) Matthew's Correlation Coefficient (MCC), which accounts for both over- and under-prediction, (3) Q_{pred} , the percentage of correct prediction of turn residues (or probability of correct prediction), and (4) Q_{obs} , the percentage of observed turn residues that are correctly predicted (or percent coverage). These measures can be calculated using the following equations:

$$Q_{total} = \left(\frac{p+n}{t} \right) \times 100 \quad Q_{pred} = \left(\frac{p}{p+o} \right) \times 100$$

$$MCC = \frac{pn - ou}{\sqrt{(p+o)(p+u)(n+o)(n+u)}} \quad Q_{obs} = \left(\frac{p}{p+u} \right) \times 100$$

where p and n are the number of correctly predicted turn and non-turn residues, respectively; o and u are the number of incorrectly predicted turn and non-turn residues, and $t = p + n + o + u$ is the total of residues.

Following the work by Kaur and Raghava,¹⁷ in addition to the four criteria mentioned above, we used a threshold independent measure, AUC (area under the curve), for the comparison. A ROC curve is obtained by plotting all *sensitivity* values (true-positive fraction) on the y -axis against their equivalent ($1 - specificity$) values (false-positive fraction) for all available thresholds on the x -axis. The AUC is taken as an important index because it provides a single measure of overall accuracy that is not dependent on a particular threshold.¹⁰ Here we used trapezoidal

intergration³ to calculate the *AUC* of ROC curves produced by our prediction methods. *Sensitivity* (Sn) and *specificity* (Sp) are defined as:

$$Sn = \frac{p}{p + u} \quad Sp = \frac{n}{n + o}$$

3. Results

We developed two support vector machine-based methods BTSVM and GTSVM. BTSVM is for predicting β -turns and analyzing the support of amino acids for the formation of β -turns. GTSVM does the same tasks for γ -turns. The settings of BTSVM and GTSVM for each task are presented in Table 2.

3.1. Prediction of β -turns and γ -turns

Table 3 shows the performance of BTSVM and 5 other methods on 426 non-homologous protein chains by sevenfold cross-validation; and Table 4 shows the performance of some methods (including ours, GTSVM) on 320 non-homologous protein chains by fivefold cross-validation. As it can be seen, BTSVM achieves a *MCC* score up to 0.43 when using PSSM and 0.45 when using additional secondary structure information, which is in turn predicted by PSIPRED; GTSVM has *MCC* of 0.11 when using PSSM and 0.13 when using additional predicted secondary structure information.

For the comparison, we set a new decision threshold for turn and non-turn classes such that Q_{pred} of our methods is (nearly) equal to that of the best methods so far (BTPRED for β -turns and SNNS for γ -turns). The accuracy of our methods at the new threshold are given in brackets in Tables 3 and 4. As can be seen, for predicting β -turns, our method BTSVM has the best performance when compared to other single methods on the criteria Q_{pred} , Q_{obs} and *MCC*, while Q_{total} is still high enough. For predicting γ -turns, although our method GTSVM gives $MCC = 0.10$, $Q_{total} = 53.0$ that are slower than those of SNNS, our $Q_{obs} = 75.9$ is significantly higher.

We also calculated the threshold independent measure *AUC* for our prediction methods by the trapezoidal method, which systematically underestimates the *AUC*.³ The *AUC* of BTSVM and GTSVM (using PSSMs) are 0.81 and 0.70 respectively (see Tables 3 and 4), which are all greater than *AUC* of previous methods

Table 2. Settings of BTSVM and GTSVM.

	Task	Parameters
BTSVM	Prediction	RBF kernel, PSSM, sliding_window_length = 12
BTSVM_LIN	Analysis	Linear kernel, PSFM, sliding_window_length = 8
GTSVM	Prediction	RBF kernel, PSSM, sliding_window_length = 5
GTSVM_LIN	Analysis	Linear kernel, PSFM, sliding_window_length = 5

Table 3. Results of β -turn/non- β -turn prediction of some methods.

		Q_{total}	Q_{pred}	Q_{obs}	MCC	AUC
Chou-Fasman	Sin. seq.	74.9 (69.3)	46.1 (36.9)	16.9 (35.3)	0.16 (0.16)	
	Sin. seq. & sec. struct.	74.3 (75.3)	47.7 (49.6)	54.3 (47.5)	0.34 (0.32)	
Thornton	Sin. seq.	74.5 (70.1)	44.0 (36.7)	16.7 (30.5)	0.15 (0.14)	
	Sin. seq. & sec. struct.	75.2 (75.2)	49.3 (49.3)	44.9 (44.9)	0.31 (0.31)	
1-4 & 2-3 correlation model	Sin. seq.	63.2 (71.1)	35.3 (40.8)	60.4 (40.3)	0.21 (0.21)	
	Sin. sec. & seq. struct.	73.4 (74.8)	46.2 (48.0)	51.5 (39.8)	0.31 (0.28)	
Sequence couple model	Sin. seq.	50.6 (72.7)	31.7 (43.9)	88.4 (41.0)	0.23 (0.25)	
	Sin. seq. & sec. struct.	72.2 (75.4)	45.0 (49.6)	60.0 (40.0)	0.33 (0.28)	
BTPRED	Sin. seq.	71.6	44.1	57.3	0.31	
	Mul. seq.	73.5	47.2	64.3	0.37	0.72
	Mul. seq. & sec. struct.	75.5	49.8	72.3	0.43	0.77
BTSVM	Sin. seq.	74.2	47.6	49.2	0.31	
	Mul. seq.	78.4 (73.4)	55.9 (47.5)	58.6 (75.4)	0.43 (0.43)	0.81
	Mul. seq. & sec. struct.	79.8 (76.0)	59.2 (50.9)	58.0 (72.0)	0.45 (0.45)	0.82
BTSVM_LIN	Mul. seq. (PSFM)	73.1	46.0	55.0	0.32	

Note: The results of Chou-Fasman, Thornton, 1-4 & 2-3 correlation model and sequence couple model at original and new (in brackets) threshold values are from (Kaur, 2002).¹⁷ The results of BTPRED are from (Kaur, 2003).¹⁹ The results of BTSVM are sevenfold cross-validation accuracies obtained in the same way. BTSVM_LIN is used for analysis of β -turns.

reported in Kaur^{18,19} (the AUC of BTPRED and SNNS using PSSMs are 0.72 and 0.69 respectively).

As in the work of Kaur and Raghava,^{18,19} we tried to take account the additional secondary structure information, which is directly predicted by the PSIPRED method¹⁵ without re-training it in the training dataset (which might be unfair for the comparison because PSIPRED might use a larger training dataset). Each protein sequence is then represented as a bi-dimensional vector $L \times 23$, where L is the length of its sequence, and each position in the protein is encoded by a group of 23 inputs, 20 units encoding for the amino acid at that position and the remaining three units being the probabilities of three states (helix, strand, and coil) provided in the output of the PSIPRED prediction. The performance of BTSVM is improved to

Table 4. Results of γ -turn/non- γ -turn prediction of some methods.

		Q_{total}	Q_{pred}	Q_{obs}	MCC	AUC
Sequence couple model	Sin. seq.	66.3	2.8	50.1	0.05	
	Sin. seq. & sec. struct.	57.8	5.9	43.2	0.08	
GOR	Sin. seq.	62.1	4.7	54.4	0.06	
	Sin. seq. & sec. struct.	75.5	6.1	45.5	0.09	
WEKA (logistic regression)	Sin. seq.	61.7	4.7	56.2	0.06	
	Mul. seq.	62.7	5.5	63.9	0.10	
	Mul. seq. & sec. struct.	62.6	5.6	65.1	0.12	
WEKA (naive Bayes)	Sin. seq.	66.5	4.8	49.5	0.06	
	Mul. seq.	59.0	5.1	65.3	0.09	
	Mul. seq. & sec. struct.	57.4	5.0	65.4	0.11	
WEKA (J48 classifier)	Sin. seq.	89.6	4.3	10.4	0.02	
	Mul. seq.	92.5	5.0	7.2	0.02	
	Mul. seq. & sec. struct.	92.6	5.0	7.2	0.03	
SNNS	Sin. seq.	56.1	4.3	59.4	0.06	
	Mul. seq.	76.6	5.1	58.6	0.12	0.69
	Mul. seq. & sec. struct.	74.0	6.3	83.2	0.17	0.73
GTSVM	Sin. seq.	61.6	4.8	57.9	0.07	
	Mul. seq.	78.7	6.9	44.5	0.11	0.70
		(53.0)	(5.1)	(75.9)	(0.10)	
	Mul. seq. & sec. struct.	79.9	7.7	47.5	0.13	0.72
		(67.4)	(6.3)	(64.7)	(0.12)	
GTSVM_LIN	Mul. seq. (PSFM)	64.7	5.4	59.3	0.09	

Note: The results of sequence couple model, GOR, SNNS, WEKA are from (Kaur, 2003).¹⁸ The results of GTSVM are fivefold cross-validation accuracies obtained in the same way. GTSVM_LIN is used for analysis of γ -turns.

$Q_{total} = 76.0$, $Q_{pred} = 50.9$, $Q_{obs} = 72.0$, $MCC = 0.45$ and $AUC = 0.82$ (Table 3) and that of GTSVM is improved to $Q_{total} = 67.4$, $Q_{pred} = 6.3$, $Q_{obs} = 64.7$, $MCC = 0.12$ and $AUC = 0.72$ (Table 4). As can be seen, BTSVM is still better than other methods, but GTSVM is worse than SNNS.

3.2. Supports of amino acid positions for the formation of β -turns and γ -turns

We used BTSVM_LIN and GTSVM_LIN with linear kernels and PSFMs (see Sec. 2.5 and Table 2) to estimate the support of amino acids at individual positions in the protein sequence (or, more briefly, the support of amino acid positions) for the formation of β -turns and γ -turns. In other words, we need to find the w_i 's in a linear SVM classification function (i.e., Eq. (2)). In this task, first we used PSFMs for BTSVM_LIN and GTSVM_LIN because PSFMs emphasize clearly the occurrence of amino acids at an individual position in protein sequence. While PSSMs (log-odds values), in addition to the information of occurrence of amino acids, take account a general substitution matrix (i.e., BLOSUM62) and other information, they might

be not as good as PSFMs in this task. We also tried to use single sequence for this task and found that the ranking of weights (w_i) is almost similar to the ranking of them generated by using PSFMs although their values are different. Here, we support that using PSFMs is more accurate because it gave a better performance (see Tables 3 and 4). Second, we chose the sliding window length of 8 for β -turns and 5 for γ -turns, because after having tried various experiments we found that these lengths make BTSVM_LIN and GTSM_LIN have the best performance.

After setting the parameters described above, we trained the BTSVM_LIN on the whole β -turn dataset B and GTSVM_LIN on the whole γ -turn dataset G to build the linear classification functions (Eq. (2)) for turn/non-turn. From these classification functions, we extracted the supports (w_i 's) of amino acid positions (see Sec. 2.5). Table 5 shows the supports of amino acids for the formation of β -turns depending on their position in the sliding window of length 8, and Table 6 shows the supports for the formation of γ -turns under the window of length 5.

In general, the support of an amino acid for the formation of β -turns/ γ -turns varies from position to position in the window. We have marked in boldface positions where certain amino acids have a strong support, and underlined positions where certain amino acids have a strong prevention.

Table 5. The supports of amino acid positions for the formation of β -turns (turn-windows) under the linear classification model of BTSVM_Lin.

Amino acid	Position 1	2	3 (i)	4 ($i + 1$)	5 ($i + 2$)	6 ($i + 3$)	7	8
Ala (A)	-0.346	<u>-0.539</u>	<u>-1.047</u>	0.223	<u>-0.622</u>	0.011	-0.435	-0.462
Arg (R)	-0.088	0.201	<u>-0.788</u>	0.349	0.275	0.271	0.377	-0.209
Asn (N)	-0.416	-0.164	0.122	0.400	2.712	0.267	0.516	-0.104
Asp (D)	-0.325	0.358	0.589	0.690	1.542	0.339	0.188	-0.367
Cys (C)	-0.131	0.138	-0.138	-0.472	0.067	0.286	0.069	0.098
Gln (Q)	-0.090	-0.227	<u>-1.140</u>	-0.083	0.106	0.594	0.122	-0.496
Glu (E)	-0.082	-0.286	<u>-1.242</u>	0.798	0.028	-0.400	0.285	-0.257
Gly (G)	-0.162	-0.044	-0.432	0.219	2.207	1.061	0.358	-0.278
His (H)	0.001	0.152	-0.412	0.250	0.641	0.499	0.382	0.186
Ile (I)	-0.094	-0.328	<u>-1.250</u>	-0.279	-0.489	<u>-0.702</u>	-0.202	-0.161
Leu (L)	-0.391	-0.311	<u>-0.877</u>	-0.299	-0.259	-0.242	0.002	-0.151
Lys (K)	-0.045	-0.221	<u>-1.021</u>	0.944	0.098	0.446	0.741	-0.211
Met (M)	-0.239	-0.312	<u>-0.738</u>	-0.279	-0.003	0.204	-0.009	-0.323
Phe (F)	-0.236	-0.150	<u>-0.648</u>	-0.409	0.368	-0.052	-0.048	-0.078
Pro (P)	0.077	-0.215	0.204	1.982	-0.254	0.263	1.234	0.227
Ser (S)	-0.206	-0.124	0.156	0.372	0.333	0.240	0.332	-0.282
Thr (T)	-0.214	-0.070	-0.427	-0.137	0.258	0.502	0.724	-0.052
Trp (W)	-0.263	-0.036	<u>-0.801</u>	-0.086	-0.044	-0.138	0.120	0.045
Tyr (Y)	0.177	0.089	<u>-0.511</u>	-0.164	0.230	-0.059	0.159	-0.125
Val (V)	0.034	0.044	<u>-0.911</u>	-0.326	<u>-0.542</u>	0.019	0.011	0.217

Note: Amino acid positions with positive supports will contribute to the formation of β -turns, others will prevent the formation of β -turns. The larger the absolute value of the support, the stronger the contribution (or prevention if negative). Amino acid positions with the strongest supports (more than 0.50) are printed in boldface. Those with the lowest supports (less than -0.50) are underlined.

Table 6. The supports of amino acid positions for the formation of γ -turns under the linear classification model of BTSVM.Lin.

Amino acid	Position 1	2 (i)	3 ($i + 1$)	4 ($i + 2$)	5
Ala	0.120	-0.960	-0.451	<u>-1.221</u>	-0.197
Arg	0.566	0.049	<u>-1.067</u>	0.581	0.287
Asn	0.693	0.280	2.508	0.162	0.879
Asp	0.733	-0.317	2.040	0.125	0.878
Cys	0.814	0.127	-0.206	0.324	-0.005
Gln	0.356	-0.473	-0.383	<u>-1.040</u>	-0.237
Glu	0.935	0.012	<u>-1.276</u>	<u>-1.517</u>	0.225
Gly	1.479	1.080	-0.691	-0.468	0.707
His	0.597	0.503	0.108	0.154	-0.130
Ile	0.609	0.250	<u>-1.024</u>	-0.559	<u>-1.102</u>
Leu	0.478	-0.066	-0.222	-0.800	-0.605
Lys	0.927	-0.518	-0.569	-0.403	-0.223
Met	0.874	0.326	1.380	0.133	-0.813
Phe	0.601	-0.182	-0.664	-0.388	-0.130
Pro	1.413	1.197	1.929	<u>-1.024</u>	1.295
Ser	1.196	-0.079	<u>-1.278</u>	0.751	0.605
Thr	0.631	-0.011	<u>-2.074</u>	0.598	0.024
Trp	0.694	-0.316	0.250	0.117	0.595
Tyr	0.402	0.411	-0.509	-0.216	-0.148
Val	0.154	-0.572	<u>-1.751</u>	0.332	0.478

Note: Amino acid positions with positive supports will contribute to the formation of γ -turns, others will prevent the formation of γ -turns. The larger the absolute value of the support, the stronger the contribution (or prevention if negative). Amino acid positions with the strongest supports (more than 1.00) are printed in boldface. Those with the lowest supports (less than -1.00) are underlined.

There are some amino acids, of course at different positions, strongly supporting both the formation of β -turns and γ -turns. For example, glycine (Gly) supports the β -turn formation at positions $i + 2$ and $i + 3$. It also supports the γ -turn formation at positions i and $i - 1$. In particular, amino acid asparagine (Asn) at position $i + 2$ has the strongest support for the formation of β -turns; and it also has the strongest support for the formation of γ -turns when it occurs at position $i + 1$. There are some amino acids, on the other hand, preventing both the formation of β -turns and γ -turns: alanine (Ala), isoleucine (Ile), etc. There are also some amino acids that, while their occurrence almost does not impact the β -turn formation (or γ -turn formation), their occurrence at specific positions strongly supports or prevents the formation of the other type of turns. For example, serine (Ser) almost does not influence the β -turn formation, but it strongly supports γ -turn formation at position $i - 1$ and strongly prevents at position $i + 1$.

4. Discussions

4.1. Prediction of β -turns and γ -turns

Our methods gave prediction results clearly and had high performance. The reasons for these may be the following:

- (1) As explained in the work of Kaur,^{18,19} our methods, like BTPRED, BetaTPred2 and GammaPred, incorporate the evolutionary information of proteins by using multiple sequence alignment. The evolutionary information has been proved to significantly improve most structure prediction methods.
- (2) Like BTPRED, BetaTPred2 and GammaPred, our methods can improve the prediction accuracy by using additional secondary structure, which is in turn predicted by a secondary structure prediction method with high accuracy, i.e., PSIPRED.
- (3) In our methods, the prediction is performed at the turn level (see Sec. 2.4). This is different from previous work (PTPRED, BetaTPred2, and GammaPred), which performed the prediction at the residue level. Therefore, all β -turns/ γ -turns predicted by our methods, containing at least four residues with a β -turn and three residues with a γ -turn, are valid and clearer. In consequence, there is no need to go through a filtering process to exclude unrealistic β -turns/ γ -turns.
- (4) Our method used SVMs, which has many advantages over neural networks. For example, it always gives the global optimal solution with a particular kernel, it is easy to control the capacity, etc.^{9,26}

4.2. Supports of amino acid positions for the formation of β -turns and γ -turns

We proposed the new term “support of an amino acid position to the formation of β -turns and γ -turns under the SVM classification model” that emphasizes the discriminative features. Our analysis results agree closely with those from previous statistical methods. That is, amino acid positions with stronger positive supports for the formation of turns are often those with the higher amino acid positional potentials (preferences) for turns as previously reported in the work of Guruprasad and Rajkumar,¹² and Chou.⁵ Conversely, amino acid positions with stronger negative supports are often those with lower amino acid positional potentials (preferences).

However, there are at least four differences between our approach and others. First, our analysis and prediction are based on the “multivariable” classification model of SVM, which is more general than previous models, such as Site-Independent model,⁸ 1-4 and 2-3 Residue-Correlation model,²⁸ and Sequence-Couple model.⁶ Therefore, the supports of amino acid positions are not considered independently, but are mutually taken by a combinational linear. This explains why the order of amino acid positions sorted by their supports is different from the order when they are sorted by their potentials (or preferences).

Second, our methods performed at the turn level by a window wider than the length of the β -turn/ γ -turn itself. Some amino acids, although may not be in turn area, have significant supports (or preventions) to the β -turn or γ -turn formation of the residues preceding or following them. This may explain why some previous statistical methods had low prediction performance, since they performed their prediction only under a window of size 4 with β -turns and 3 with γ -turns.

Third, as explained above, our approach emphasizes the discriminative features due to the discriminative character of SVM model.

Fourth, the analysis results of our approach are more comprehensive and therefore easier to use than those of others. Amino acid positions with positive supports will contribute to the formation of turns; otherwise they will prevent. The stronger the support, the stronger the affect of the amino acid position on the formation of turns.

Acknowledgements

This work was supported by a Grant-in-Aid for Scientific Research on Priority Areas (C) "Genome Information Science" from the Ministry of Education, Culture, Sports, Science, and Technology of Japan; BIRD of Japan Science and Technology Agency (JST); and the COE project JCP KS1 from Japan Advanced Institute of Science and Technology. We would like to thank Harpreet Kaur for providing us some important information about BTPRED. We are grateful to Chih Jen Lin for providing the software LIBSVM and some email discussion about it. We also thank to Jose C. Clemente and anonymous reviewers for their criticism and suggestions of the reading of the manuscript.

References

1. Altschul S, Koonin E, Iterated profile searches with psi-blast — a tool for discovery in protein databases, *Trends Biochem Sci* **23**:444–447, 1998.
2. Altschul S, Madden T, Shaffer A, Zhang J, Zhang Z et al., Gapped blast and psi-blast: a new generation of protein database search programs, *Nucl Acids Res* **25**:3389–3402, 1997.
3. Bradley AD, The use of the area under the roc curve in the evaluation of machine learning algorithms, *Pattern Recognition* **30**(7):1145–1159, 1997.
4. Brank J, Grobelnik M, Milic-Frayling N, Mladenic D, Feature selection using support vector machines, in *3rd International Conference on Data Mining Methods and Databases for Engineering, Finance and Other Fields*, pp. 261–273, 2002.
5. Chou KC, Prediction of tight turns and their types in proteins, *Analytical Biochem* **286**:1–16, 2000.
6. Chou KC, Blinn JR, Classification and prediction of β -turn types, *J Protein Chem* **16**:575–595, 1997.
7. Chou PY, Fasman GD, Conformational parameters for amino acids in helical, β -sheet and random coil regions calculated from proteins, *Biochemistry* **13**:211–222, 1974.
8. Chou PY, Fasman GD, Prediction of β -turns, *Biophys J* **26**:367–384, 1979.
9. Cristianini N, Shawe Taylor J, *An Introduction to Support Vector Machines*. Cambridge, 2002.
10. Deleo J, Measuring classifier intelligence, in *Second International Symposium on Uncertainty Modelling and Analysis*, pp. 318–325, 1993.
11. Gribskov M, McLachlan AD, Eisenberg D, Profile analysis: detection of distantly related proteins, *PNAS* **84**(13):4355–4358, 1987.
12. Guruprasad K, Rajkumar S, β - and γ -turns in proteins revisited: a new set of amino acid dependent positional preferences and potential, *J Biosci* **25**:143–156, 2000.

13. Guyon I, Weston J, Barnhill S, Vapnik V, Selection for cancer classification using support vector machines, *Machine Learning* **46**(1/3):389, 2002.
14. Hutchinson EG, Thornton JM, A program to identify and analyze structural motifs in proteins, *Protein Sci* **5**:212–220, 1996.
15. Jones DT, Protein secondary structure prediction based on position-specific scoring matrices, *Mol Biol* **292**:195–202, 1999.
16. Kai YD, Liu XJ, Xu XB, Chou KC, Support vector machines for the classification and prediction of β -turn types, *J Pep Sci* **8**:297–301, 2002.
17. Kaur H, Raghava GPS, An evaluation of β -turn prediction methods, *Bioinformatics* **18**:1508–1514, 2002.
18. Kaur H, Raghava GPS, A neural network based method for prediction of gamma-turns in proteins from multiple sequence alignment, *Protein Sci* **12**:923–929, 2003.
19. Kaur H, Raghava GPS, Prediction of β -turns in proteins from multiple alignment using neural network, *Protein Sci* **12**:627–634, 2003.
20. Kim H, Park H, Protein secondary structure prediction by support vector machines and position-specific scoring matrices, *Protein Engin* 2003.
21. Minakuchi Y, Satou K, Konagaya A, Prediction of protein-protein interaction sites using support vector machines, in *International Conference on Mathematics and Engineering Techniques in Medicine and Biological Sciences*, pp. 22–28, 2003.
22. Richardson JS, The anatomy and taxonomy of protein structure, *Adv Protein Chem* **34**:167–339, 1981.
23. Rose GD, Gierasch LM, Smith JA, Turns in peptides and proteins, *Adv Protein Chem* **37**:100–109, 1985.
24. Shepherd AJ, Gorse D, Thornton JM, Prediction of the location and type of β -turns in proteins using neural networks, *Protein Sci* **8**:1045–1055, 1999.
25. Takano K, Yamagata Y, Yutani K, Role of amino acid residues at turns in the conformational stability and folding of human lysozyme, *Biochemistry* **39**:8655–8665, 2000.
26. Vapnik V, *Statistical Learning Theory*, Wiley N.Y., 1998.
27. Wilmot CM, Thornton JM, Analysis and prediction of the different types of β -turns in proteins, *J Mol Biol* **203**:221–232, 1988.
28. Zhang CT, Chou KC, Prediction of β -turns in proteins by 1-4 & 2-3 correlation model, *Biopolymers* **41**:673–702, 1997.



Tho Hoan Pham currently is a Ph.D. candidate in School of Knowledge Science, Japan Advanced Institute of Science and Technology, Japan. He received the B.Sc. degree in mathematics from Hanoi University of Education, Vietnam in 1993 and the M.Sc. degree in computer science from Hanoi University of Technology, Vietnam in 1997. He is interested in machine learning approach to bioinformatics.



Kenji Satou is an associate professor of School of Knowledge Science, Japan Advanced Institute of Science and Technology. He received the B.E. and M.E. degrees in computer science from Kyushu University in 1987 and 1989, respectively, and Ph.D. in computer science from Kyushu University in 1996. His current research interests include various topics in bioinformatics. Especially, molecular interaction analysis, literature analysis, and Grid computing are his favorite.



Tu Bao Ho is a professor of School of Knowledge Science, Japan Advanced Institute of Science and Technology. He received the B.Tech. degree in applied mathematics from Hanoi University of Technology (1978), M.S. and Ph.D. degrees in Computer Science from Pierre and Marie Curie University, Paris (1984, 1987), and Habilitation from Paris Dauphine University (1998). His current research interests include knowledge-based systems, machine learning, bioinformatics, knowledge discovery and data mining.