Prediction of T-Cell Epitopes Using Biosupport Vector Machines

Zheng Rong Yang* and Felicia Charles Johnson

Department of Computer Science, University of Exeter, United Kingdom

Received January 4, 2005

The immune system is concerned with the recognition and disposal of foreign or "non self" molecules or cells that enter the body of an immunologically competent individual. The generation of an immune response depends on the interaction of components, namely, the immunogen (nonself or foreign cell or molecule), antibody producing humoral immune system, and sensitized lymphocyte producing cellular immune system. An immunogen possesses surface structures referred to as epitopes; the precise pattern of each epitope enables an individual's immune system to recognize cells or molecules as self or immunogens. During the recognition process, the specific cells known as macrophages identify the epitope structures on the immunogen and save them in the form of short peptides 10−18 amino-acids-long known as immune dominant peptides (IDPs). IDPs are then bound with surface proteins on macrophages known as MHC protein complexes. The macrophages then present this IDP−MHC complex to a T cell that possesses a specific receptor that is specific for the foreign epitope on the IDP bound to MHC complex. This initiates an immune system cascade that results in the disposal of the immunogen. The study and accurate prediction of T-cell epitopes is, thus, very important for designing vaccines against pathogenic diseases. The present study applied the newly developed biosupport vector machine to the T-cell epitope data. This new algorithm introduces a biobasis function into the conventional support vector machines so that the nonnumerical attributes (amino acids) in protein sequences can be recognized without a feature extraction process, which often fails to properly code the biological content in protein sequences. The prediction accuracy of a 10-fold cross validation is 90.31%, compared with 87.86% using support vector machines reported as the best compared with other algorithms in an earlier study.

INTRODUCTION

The most important capability of the immune system is the recognition between host (self) and foreign molecules. Any foreign molecule is commonly regarded as a potential pathogen. The recognition of these foreign proteins is completed by the T cells, which can discriminate molecules binding to the major histocompatibility complex (MHC) receptors against the native ones.[1] In terms of this, the T cells are the key components in regulating a specific immune response.[2] The MHC molecule plays a role in binding and presenting IDPs on the macrophage surface for recognition by epitope-specific T-cell receptors (TCRs) of T lymphocytes.[3] The short peptides presented to TCRs are referred to as T-cell epitopes, which are the important components for studying the specificity of the immune system. The accurate prediction of these T-cell epitopes is, therefore, a critical step toward the efficient design of vaccines and immunotherapies.[3] The study of T-cell epitopes is of paramount importance for finding cures against diseases such as chronic Lyme disease[4] and the hepatitis C virus infection.[5]

The use of computer programs for the prediction of T-cell epitopes can complement laboratory experiments to increase the efficiency of T-cell epitope screening. The combination of accurate predictors with new experimental methods for the recognition of T-cell epitopes will allow tracking of antigen-specific responses in clinical studies.[6] Therefore, the accurate prediction of T-cell epitopes is of particular importance in clinical immunology and vaccine design. Many computational methods have been used for the prediction of T-cell epitopes.[6,7] For instance, various profile matrices were derived on the basis of the experimentally determined peptides for the prediction process,[8−10] and binding motifs were also used for this task.[11] A quantitative method was used to predict the relative binding strengths of all possible nonamer peptides to the MHC class I molecule HLA-A2 on the basis of experimental peptide binding data.[12] Statistical methods such as the partial least-squares-based multivariate statistical approach were also used for this task.[13] In addition, some state-of-the-art computational algorithms have been paid attention to recently; for instance, artificial neural networks (ANNs),[1,6,14] hidden Markov models (HMMs),[6,15] decision trees,[16] and support vector machines (SVMs)[17] have been used. SVMs certainly outperform the other computational algorithms because SVMs are able to maximize the generalization capability of a trained predictor. This may not be easily achieved using HMMs or ANNs because, as indicated by Yu et al.,[6] ANNs are prone to having a higher specificity and a lower sensitivity and HMMs are prone to having a higher sensitivity and a lower specificity. Other than the application to the prediction of T-cell epitopes, SVMs have also been applied to the prediction of translation initiation sites,[18] the classification of proteins,[19] the prediction of the alpha and beta turns,[20] the prediction of phosphorylation sites,[21] and the prediction of protein−protein interac-

* Corresponding author e-mail: z.r.yang@ex.ac.uk or z.r.yang@exeter.ac.uk.

10.1021/ci050004t CCC: $30.25
Published on Web 06/29/2005

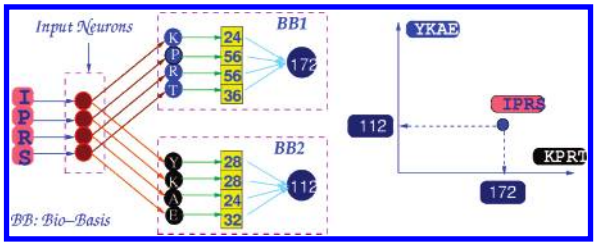

Figure 1. Illustration of the biomap.

tions.[22] More research on protein−protein interaction,[23] nonlinear dynamics of biological neurons[24] and gene selection,[25] and many others[26] have been paid attention to. A review of the biological applications of SVMs can be seen in ref 27.

As amino acids in peptides are nonnumerical attributes, an encoding process must be completed, and the most popular method is the distributed encoding method.[28] However, the distributed encoding method has two major problems, that is, the enlargement of the model parameters and the difficulty with encoding biological content in peptides.[29,30] On the basis of this understanding, the biosupport vector machine (bSVM) was proposed.[31] This paper is aimed to investigate if bSVM will outperform SVMs so as to improve the accuracy of predicting T-cell epitopes.

METHODS

Data. A total of 203 10-mer T-cell epitope peptides were downloaded from http://linus.nci.nih.gov/Data/LAU203_Peptide.pdf.[17] The data set is composed of 167 nonepitopes and 36 epitopes.

Biosupport Vector Machine. The bSVM was proposed by Yang and Chou.[31] The working principle of bSVM is the same as that of SVMs.[32] When using bSVM for modeling *k*-mer peptides, a kernel function needs to be specially designed. The kernel function used in bSVM is called the biobasis function, first used in the biobasis function neural network.[29] The biobasis function method has been successfully applied to some functional site prediction tasks, for instance, the prediction of trypsin cleavage sites,[29] HIV cleavage sites,[30] hepatitis C virus protease cleavage sites,[33] disordered proteins,[34] phosphorylation sites,[35] O-linkage sites in glycoproteins,[36] and caspase cleavage sites.[37]

Each biobasis is supported by a *k*-mer peptide referred to as a template peptide. The relation between a novel *k*-mer peptide and a template peptide is quantified as a homology alignment score using a mutation matrix like the Dayhoff matrix.[38,39] The homology alignment score is further normalized. For instance, in Figure 1, a query 4-mer peptide (IPRS) will be aligned with two 4-mer template peptides (KPRT and YKAE) to produce two homology alignment scores (24 + 56 + 56 + 36 = 172 and 28 + 28 + 24 + 32 = 112, respectively). The values (24, 56, 36, 28, 24, and 32) are obtained from the Dayhoff matrix. Because 172 > 112, it is believed that the query peptide shares more functional similarity with the first template peptide. This process is called the biomap.

The same as SVMs, bSVM also selects the *k*-mer peptides that are most difficult to classify as the support peptides. In Figure 2, there are six 4-mer support peptides, that is, YAKE, TGGA, AWCV, KPRT, TFGH, and PKRA.

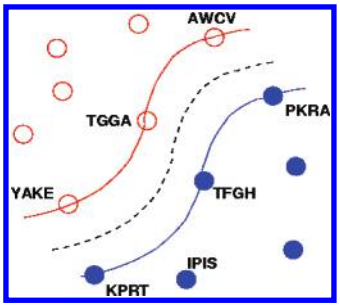

Figure 2. Demonstration of the support peptides. Two classes of peptides are represented using the filled and open circles. The dashed curve represents the hyperplane for separation, whereas the thick curves represent the boundary of the margin. As the query peptide IPIS is close to the support peptide KPRT, it is located in the area occupied by the filled circles.

The bSVM classifier using these six support peptides can be defined as

$$y(x) = w_{\mathrm{YAKE}}\,\phi(x,\mathrm{YAKE}) + w_{\mathrm{TGGA}}\,\phi(x,\mathrm{TGGA}) + w_{\mathrm{AWCV}}\,\phi(x,\mathrm{AWCV}) + w_{\mathrm{KPRT}}\,\phi(x,\mathrm{KPRT}) + w_{\mathrm{TFGH}}\,\phi(x,\mathrm{TFGH}) + w_{\mathrm{PKRA}}\,\phi(x,\mathrm{PKRA}) \quad (1)$$

where $\phi(...)$ is the kernel function, w is the weight of the indicated peptide, x is the query peptide, and $y(x)$ is the prediction of x. Suppose the set of K template peptides is denoted as $\Theta = \{t_1, t_2, ..., t_K\}$, then the kernel function is defined as

$$\phi(x,\mathrm{YAKE}) = \phi[f(x,\Theta), f(\mathrm{YAKE},\Theta)] \quad (2)$$

where $f(x,\Theta) = [f(x,t_1), f(x,t_2), ..., f(x,t_K)]$ and $f(...)$ is the biobasis function, which has been defined in Thomson et al.[29] Additionally,

$$f(x,t_K) = \exp\left[-\alpha\frac{M(x,t_K) - M(t_K,t_K)}{M(t_K,t_K)}\right] \quad (3)$$

where α is a positive constant (10 in this paper) and

$$M(x,t_K) = \ln\prod_{d=1}^{D} p(x_d,t_{Kd}) = \sum_{d=1}^{D} m(x_d,t_{Kd}) \quad (4)$$

The value of $m(x_d,t_{Kd})$ can be obtained from a mutation matrix like the Dayhoff matrix.[34] Note that $x_d,t_{Kd} \in A$ and A is the set of 20 amino acids.

Procedure. *Step 1*. A total of 203 10-mer peptides are randomly divided into 10 nonoverlapping folds for 10-fold cross validation.

Step 2. Map nine folds of the 10-mer peptides into a high-dimensional numerical space using the biobasis function. There are two strategies:

Strategy 1. Both positive and negative 10-mer peptides are the candidates for template peptide selection. The model constructed this way is referred to as bSVM1.

Strategy 2. Only positive 10-mer peptides are the candidates for template peptide selection. The use of this strategy is based on the observation that negative peptides generally do not have conserved patterns.[35] The model constructed this way is referred to as bSVM2. The method of selecting template peptides will be discussed in the Discussion section.

Step 3. The prediction of a query peptide is based on the Bayes rule

$$p(C|x) = \frac{p(x|\mathrm{C})\, p(\mathrm{C})}{p(x)} \tag{5}$$

where $p(x|\mathrm{C})$ is the conditional probability that x belongs to class C (either T-cell epitope or not), $p(C)$ is the prior probability of class C, $p(x)$ is the normalizing factor, and $p(C|x)$ is the posterior probability by which a decision is made. The conditional probability density function is estimated in a nonparametric method. The reason for doing so will be described in the Discussion section. In testing, a cost function is commonly introduced. This means that the final decision is made by

$$\underset{C}{\mathrm{argmax}}\{\alpha_C p(C|x)\} \tag{6}$$

where α_C is the cost function associated with class C. The final annotation of a query peptide depends on the magnitudes of $\alpha_T p(T|x)$ and $\alpha_N p(N|x)$, where α_T and α_N are the cost functions of T-cell and non-T-cell epitopes, respectively, and $p(T|x)$ and $p(N|x)$ are the respective posterior probabilities of the two classes. Note that $\alpha_T + \alpha_N = 1$.

Steps 2 and 3 are repeated 10 times.

Step 4. The prediction performance is assessed using six indicators, the true positive fraction (TPf), the true negative fraction (TNf), the total accuracy (Total), the Matthews' correlation coefficient (MCC),[40] the positive prediction power (PPf), and the receiver operating characteristic (ROC) curve.[41] Let TN, TP, FN, and FP denote true negative, true positive, false negative, and false positive, respectively. Note that positive means T-cell epitope class. The definitions of the first five indicators are as follows:

$$\mathrm{TNf} = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}}$$

$$\mathrm{TPf} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}$$

$$\mathrm{PPf} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}$$

$$\mathrm{Total} = \frac{\mathrm{TN} + \mathrm{TP}}{\mathrm{TN} + \mathrm{FP} + \mathrm{TP} + \mathrm{FN}}$$

$$\mathrm{MCC} = \frac{\mathrm{TP} \times \mathrm{TN} - \mathrm{FP} \times \mathrm{FN}}{\sqrt{(\mathrm{TN} + \mathrm{FN})(\mathrm{TN} + \mathrm{FP})(\mathrm{TP} + \mathrm{FN})(\mathrm{TP} + \mathrm{FP})}} \tag{7}$$

The positive prediction power measures the likelihood that a predicted positive is the true positive. The Matthews' correlation coefficient measures how the predictions correlate with the real target values. If the coefficient is positive, the predictions are correlated with the target values; otherwise, they are uncorrelated. The larger the value, the better the prediction performance is.

When using the ROC curve for analysis, a two-dimensional space is formulated using TPf as the vertical axis and FPf as the horizontal one. For any cost function α_T ($\alpha_N = 1 - \alpha_T$), the model will have a specific TPf value and a specific FPf value. These two values locate the model a unique point in the two-dimensional space. Changing the α_T value will then generate a series of points in this two-dimensional space. Connecting all these points produces a ROC curve. Because a model with a comparatively large TPf for a fixed FPf will be preferred, the larger the area under the ROC curve, the better the performance a classifier has. This means that we can select a robust model through maximizing the area under the ROC curves (AUR).

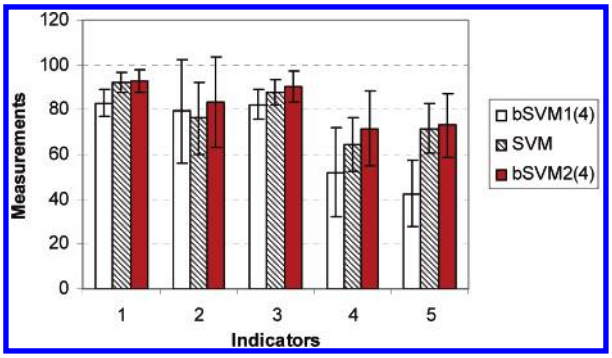

Figure 3. Indicators for three models, i.e., the SVM,[17] bSVM1, and bSVM2(4) models. The horizontal axis represents the indicators, whereas the vertical axis represents the measurements taken on three models for these indicators. The standard deviation is shown as well.

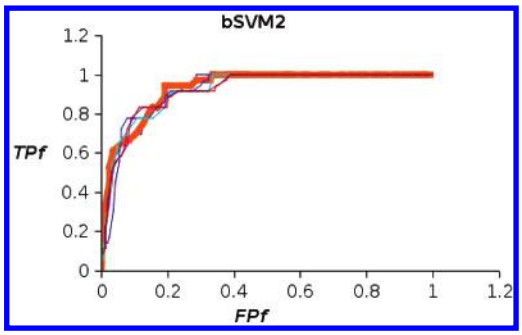

Figure 4. ROC curves for the bSVM2 models with the polynomial orders 2, 3, 4, 5, 6, and 7. The horizontal axis represents the false positive fraction (FPf), whereas the vertical axis represents the true positive fraction (TPf). The thickest line represents the model with the polynomial order as 4.

RESULTS

Shown in Figure 3 is the comparison between the SVM, bSVM1(4), and bSVM2(4) models, where the number within the parentheses is the power order of a polynomial kernel function, $[f(x,\Theta)\cdot f(\mathrm{YAKE},\Theta) + c]^k$. Note that $c = 0$. It has been found that the polynomial kernel function with the power order (k) equal to 4 performed the best, and the polynomial kernel function outperformed all the other kernel functions. From Figure 3, it can be seen that bSVM2(4) outperformed SVM and bSVM1(4) for all five indictors., that is, TNf, TPf, Total, MCC, and PPf. As there is no record of a ROC analysis in Zhao et al.,[17] the comparison of ROC analyses is omitted here. The mean and standard deviation values of TNf, TPf, and Total for the SVM models were 92.24% (4.25%), 76.21% (16.29%), and 87.86% (5.78%) respectively,[17] whereas they are 93.06% (4.69%), 83.29% (20.22%), and 90.31% (6.78%) for the bSVM2(4) models. Note that the percentage values within the parentheses are the standard deviations.

Shown in Figure 4 are the ROC curves for the bSVM2(k) models, where k varies from 2 to 7. The thickest line represents the bSVM2(4) models.

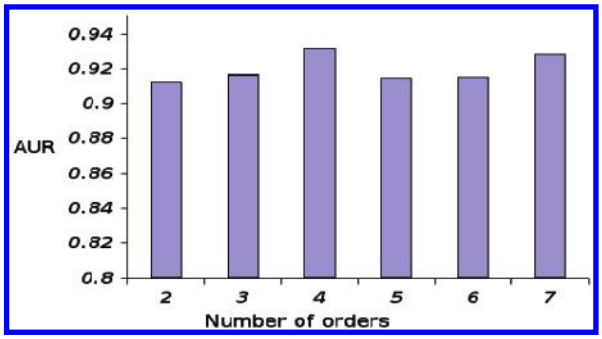

Figure 5. Areas under the ROC curves for the bSVM2 models with the polynomial orders 2, 3, 4, 5, 6, and 7. The horizontal axis represents the number of the orders, whereas the vertical one represents the AUR values. It can be seen that the model with the polynomial order as 4 demonstrated the highest AUR value.

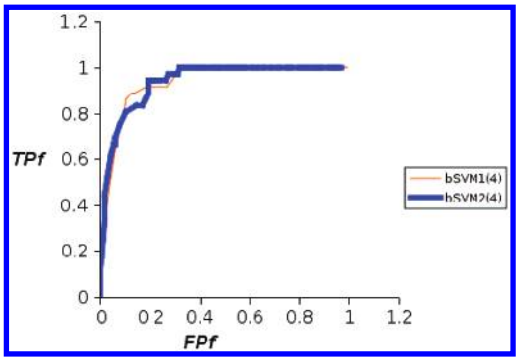

Figure 6. ROC curve comparison between the bSVM1(4) and bSVM2(4) models.

Shown in Figure 5 is the comparison of the AURs. It clearly indicates that bSVM2(4) outperformed all the others. The AUR value for the bSVM2(4) model is 0.931.

Shown in Figure 6 are the ROC curves of the bSVM1(4) and bSVM2(4) models. The AUR values are 0.931 and 0.928 for the bSVM2(4) and bSVM1(4) models, respectively.

DISCUSSION AND CONCLUSION

In each run of cross validation, nine folds of peptides are randomly divided into two parts. One is for template peptides and the other for training.

Note that the decision making in using bSVMs is based on a nonparametric method. Figure 7 shows the probability density function of one bSVM model. It has been found that they do not follow normal distributions, particularly the negative data. In addition, the threshold is biased toward the left rather than at the position of zero. This then urges the use of a nonparametric method for estimating the conditional probability density functions. With a nonparametric method, all the training predictions, each of which is assumed to be located in the center of a normal distribution, are stored. If we denote a training prediction of the cth class and nth training peptide as $\hat{y}_{cn}$, the nonparametric estimate of the conditional probability density function is as follows:

$$p(x|C) = \frac{1}{N_C}\sum_{n=1}^{N_C}\frac{1}{\sqrt{2\pi\sigma^2}}\exp\left\{-\frac{[\hat{y}(x)-\hat{y}_{cn}]^2}{2\sigma^2}\right\} \quad (8)$$

where σ is the mean distance between $\hat{y}_{cn}$'s and N_C is the number of peptides in class C. The final decision is completed using the Bayes rule as mentioned above on the basis of the prior probabilities (equal for two classes in this paper) and cost functions (equal for two classes in this study).

Currently, each residue in the sequences plays an equal role in using the biobasis function. It is known that the amino acids at sites P_5 and P_6 are important for binding. This will be dealt with by revising the mutation matrix used by the biobasis function, and a new optimization algorithm should be proposed. This will be our future work.

It should be noted that the increase on the total prediction accuracy using the newly developed bSVM is only 2.45%. This is because there is a little increase in the prediction accuracy of nonepitopes (0.82%) although the increase on the prediction accuracy of epitopes is 7.08%. We have noticed that the standard deviation of the prediction accuracy of the nonepitopes is much smaller than that of the epitopes. This can be interpreted as the diversity among the collected epitopes being much higher than that of the collected nonepitopes. Because of this, the gain of using the biobasis function, which is able to effectively code the biological content in sequences, is little on nonepitopes and large on epitopes. Because the percentage of the epitopes in the data set is only 18%, the total increase in the prediction accuracy is small, although the increase on the prediction accuracy of the epitopes is significant.

This paper has presented a method of using bSVM for the prediction of the T-cell epitopes. It shows that bSVM models outperformed the SVM models. Moreover, using only positive peptides as template peptides can enhance the performance further. It has also been found that the polynomial kernel function outperformed all the other kernel functions. In addition, the prediction capability is optimized when the power order is 4.

The programs were encoded in Java and C on a PC containing an 850 MHz Pentium and using a Linux operating system. The SVM*light* [42] package was used. The Java program

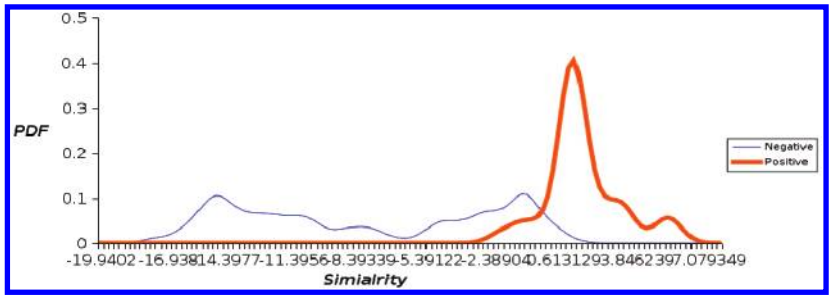

Figure 7. Probability density of the output values from one SVM model. The horizontal axis represents the output value, whereas the vertical axis represents the probability density value. The thin line represents the negative inputs, whereas the thick line represents the positive ones.

of the biomap used in bSVM can be obtained by request to the author.

ACKNOWLEDGMENT

The author thanks Dr. T. Joachims for the use of the SVM^{light} package and Dr. Zhao for providing the data.

REFERENCES AND NOTES

(1) Nielsen, M.; Lundegaard, C.; Worning, P.; Lauemoller, S. L.; Lamberth, K.; Buss, S.; Brukak, S.; Lund, O. Reliable prediction of T-cell epitopes using neural networks with novel sequence representations. *Protein Sci.* **2003**, *12*, 1007−1017.

(2) Donnes, P.; Elofsson, A. Prediction of MHC class I binding peptides, using SVMHC. *BMC Bioinf.* **2002**, *3*, 1−8.

(3) Srinivasan, K. N.; Zhang, G. L.; Khan, A. M.; August, J. T.; Brusic, V. Prediction of class I T-cell epitopes: evidence of presence of immunological hot spots in sice antigenes. *Bioinformatics* **2004**, *20*, s297−s302.

(4) Hemmer, B.; Gran, B.; Zhao, Y.; Marques, A.; Pascal, J.; Tzou, A.; Kondo, T.; Cortese, I.; Bielekova, B.; Straus, S. E.; McFarland, H. F.; Houghten, R.; Simon, R.; Pinilla, C.; Martin, R. Identification of candidate T-cell epitopes and molecular mimics in chronic Lyme disease. *Nat. Med. (NY, NY, U.S.)* **1999**, *5*, 1375−1382.

(5) Guo, H.; Yin, Y.; Wang, W.; Zhang, C.; Wang, T.; Wang, Z.; Zhang, J.; Cheng, H.; Wang, H. Sequence evolution of putative cytotoxic T cell epitopes in NS3 region of hepatitis C virus. *World J. Gastroenterol* **2004**, *10*, 847−851.

(6) Yu, K.; Petrovsky, N.; Schonbach, C.; Koh, J. Y.; Brusic, V. Methods for prediction of peptide binding to MHC molecules: a comparative study. *Mol. Med.* **2002**, *8*, 137−148.

(7) Schirle, M.; Weinschenk, T.; Stevanovic, S. Combining computer algorithms with experimental approaches permits the rapid and accurate identification of T cell epitopes from defined antigens. *J. Immunol. Methods* **2001**, *257*, 1−16.

(8) Schönbach, C.; Ibe, M.; Shiga, H. Fine-tuning of peptide binding to HLA-B*3501 molecules by nonanchor residues. *J. Immunol.* **1995**, *154*, 5951−5958.

(9) Mallios, R. R. Class II MHC quantitative binding motifs derived from a large molecular database with a versatile iterative stepwise discriminant analysis meta-algorithm. *Bioinformatics* **1999**, *15*, 432−439.

(10) Rammensee, H.; Bachmann, J.; Emmerich, N. P.; Bachor, O. A.; Stevanovic, S. SYFPEITHI: database for MHC ligands and peptide motifs. *Immunogenetics* **1999**, *50*, 213-219.

(11) Rammensee, H. G.; Friede, T.; Stevanoviic, S. MHC ligands and peptide motifs: first listing. *Immunogenetics* **1995**, *41*, 178−228.

(12) Parker, K. C.; Bednarek, M. A.; Coligan, J. E. Scheme for ranking potential HLA-A2 binding peptides based on independent binding of individual peptide side-chains. *J. Immunol.* **1994**, *152*, 163−175.

(13) Guan, P.; Doytchinova, I. A.; Zygouri, C.; Flower, D. R. MHCPred: a server for quantitative prediction of peptide-MHC binding. *Nucleic Acids Res.* **2003**, *31*, 3621−3624.

(14) Honeyman, M. C.; Brusic, V.; Stone, N. L.; Harrison, L. C. Neural network-based prediction of candidate T-cell epitopes. *Nat. Biotechnol.* **1998**, *16*, 966−969.

(15) Mamitsuka, H. Predicting peptides that bind to MHC molecules using supervised learning of hidden Markov models. *Proteins* **1998**, *33*, 460−474.

(16) Savoie, C. J.; Kamikawaji, N.; Sasazuki, T.; Kuhara, S. Use of BONSAI decision trees for the identification of potential MHC class I peptide epitope motifs. *Pac. Symp Biocomput.* **1999**, 182−189.

(17) Zhao, Y.; Pinilla, C.; Valmori, D.; Martin, R.; Simon, R. Application of support vector machines for T-cell epitopes prediction. *Bioinformatics* **2003**, *19*, 178−1984.

(18) Zien, A.; Ratsch, G.; Mika, S.; Scholkopf, B.; Lengauer, T.; Muller, K. R. Engineering support vector machine kernels that recognize translation initiation sites. *Bioinformatics* **2000**, *16*, 799−807.

(19) Zavaljevski, N.; Stevens, F. J.; Reifman, J. Support vector machines with selective kernel scaling for protein classification and identification of key amino acid positions. *Bioinformatics* **2002**, *18*, 689−696.

(20) Cai, Y. D.; Feng, K. Y.; Li, Y. X.; Chou, K. C. Support vector machine for predicting alpha-turn types. *Peptides* **2003**, *24*, 629−630.

(21) Kim, J. H.; Lee, J.; Oh, B.; Kimm, K.; Koh, I. Prediction of phosphorylation sites using SVMs. *Bioinformatics* **2005**, in press.

(22) Koike, A.; Takagi, T. Prediction of protein−protein interaction sites using support vector machines. *Protein Eng., Des. Sel.* **2004**, *17*, 165−73.

(23) Dohkan, S.; Koike, A.; Takagi, T. Prediction of protein−protein interactions using support vector machines. *IEEE 4th Symposium on Bioinformatics and Bioengineering*, 19−21 May, 2004; pp 576−583.

(24) Frontzek, T.; Navin Lal, T.; Eckmiller, R. Predicting the nonlinear dynamics of biological neurons using support vector machines with different kernels. *IEEE International Joint Conference on Neural Networks*, 15−19 July, 2001; pp 1492−1497.

(25) Chu, F.; Wang, L. Gene expression data analysis using support vector machine. *Bioinformatics using Computational Intelligence Paradigms*; Seiffert, U., Jain, L. C., Eds.; Springer: New York, 2005; pp 167−189.

(26) Wang, L. *Support Vector Machines: Theory and Applications*; Springer: New York, 2005.

(27) Yang, Z. R. Biological applications of support vector machines. *Briefings Bioinf.* **2004**, *5*, 328−338.

(28) Qian, N.; Sejnowski, T. J. Predicting the secondary structure of globular proteins using neural network models. *J. Mol. Biol.* **1988**, *202*, 865−884.

(29) Thomson, R.; Hodgman, C. T.; Yang, Z. R.; Doyle, A. K. Characterising Proteolytic Cleavage Site Activity Using Bio-Basis Function Neural Network. *Bioinformatics* **2003**, *19*, 1741−1747.

(30) Yang, Z. R.; Thomson, R. A novel neural network method in mining molecular sequence data. *IEEE Trans. Neural Networks* **2005**, *16*, 263−274.

(31) Yang, Z. R.; Chou, K. C. Bio-support vector machines for computational proteomics. *Bioinformatics* **2004**, *20*, 735−741.

(32) Vapnik, V. *The Nature of Statistical Learning Theory*; Springer-Verlag: New York, 1995.

(33) Yang, Z. R.; Berry, E. Reduced bio-basis function neural networks for protease cleavage site prediction. *J. Comput. Biol. Bioinf.* **2004**, *2*, 511−531.

(34) Thomson, R.; Esnouf, R. Predict disordered proteins using bio-basis function neural networks. *Lect. Notes Comput. Sci.* **2004**, *3177*, 19−27.

(35) Berry, E.; Dalby, A.; Yang, Z. R. Reduced bio basis function neural network for identification of protein phosphorylation sites: Comparison with pattern recognition algorithms. *Comput. Biol. Chem.* **2004**, *28*, 75−85.

(36) Yang, Z. R.; Chou, K. C. Predicting the O-linkage sites in glycoproteins using bio-basis function neural networks. *Bioinformatics* **2004**, *20*, 903−908.

(37) Yang, Z. R. Prediction of Caspase Cleavage Sites Using Bayesian Bio-Basis Function Neural Networks. *Bioinformatics* **2005**, in press.

(38) Dayhoff, M. O.; Schwartz, R. M.; Orcutt, B. C. A model of evolutionary change in proteins. Matrices for detecting distant relationships. In *Atlas of protein sequence and structure*; Dayhoff, M. O., Ed.; National Biomedical Research Foundation: Washington, DC, 1978; Vol. 5, pp 345−358.

(39) Johnson, M. S.; Overington, J. P. A structural basis for sequence comparisons—an evaluation of scoring methodologies. *J. Mol. Biol.* **1993**, *233*, 716−738.

(40) Matthews, B. W. Comparison of the predicted and observed secondary structure of T4 phage lysozyme. *Biochim. Biophys. Acta* **1975**, *405*, 442−51.

(41) Metz, C. E. Basic principles of ROC analysis. *Semin. Nucl. Med.* **1978**, *8*, 283−298.

(42) Joachims, T. Making large-scale SVM learning practical. In *Advances in Kernel Methods − Support Vector Learning*; Scholkopf, B., Burges, C., Eds.: MIT Press: Cambridge, MA, 1999.

CI050004T