Journal of the Korean Physical Society, Vol. 44, No. 3, March 2004, pp. 599∼604

Profile-Based Nearest Neighbor Method for Pattern Recognition

Keehyoung Joo
School of Computational Sciences, Korea Institute for Advanced Study, Seoul 130-722 and Department of Physics and Institute of Basic Science, SungKyunKwan University, Suwon 440-746

Julian Lee
School of Computational Sciences, Korea Institute for Advanced Study, Seoul 130-722 Department of Bioinformatics and Life Science, Soongsil University, Seoul 156-743 and Bioinformatics and Molecular Design Technology Innovation Center, Soongsil University, Seoul 156-743

Seung-Yeon Kim, Ilsoo Kim and Jooyoung Lee*
School of Computational Sciences, Korea Institute for Advanced Study, Seoul 130-722

Sung Jong Lee
School of Computational Sciences, Korea Institute for Advanced Study, Seoul 130-722 and Department of Physics and Center for Smart Bio-Materials, The University of Suwon, Whasung-kun, Kyunggi-do 445-743

(Received 8 October 2003)

We propose a nearest neighbor method of pattern recognition which is based on a weighted distance measure between patterns derived from profiles. There are a few new ingredients to the proposed method, compared to the conventional nearest neighbor methods. The distance measure is defined as a weighted sum of each pattern component, and the weight parameters are optimized. We introduce a second-layer prediction procedure analogous to that in neural network methods. We first construct a pattern database, where the classification of each pattern is already known. Prediction for a query pattern is performed by examining patterns close to it. We apply the proposed method to predict the protein secondary structure of the proteins in the CB513 set and 29 proteins from CASP5 in blind fashion. We find that the performance of our approach, especially with the second-layer prediction, is almost comparable to the state-of-the-art method based on neural network methods.

PACS numbers: 05.10.−a, 42.30.Sy, 89.75.Kd, 87.14.Ee
Keywords: Pattern recognition, Nearest neighbor method, Protein secondary structure, Pattern database

I. INTRODUCTION

Pattern recognition is an important problem in science and technology, with applications to a wide range of fields including data classification, image analysis, speech analysis, *etc.* [1, 2]. It is also very important in bioinformatics, where a huge amount of biological information is to be analyzed. Various algorithms such as nearest neighbor methods [3], Bayesian statistics methods [4] and neural network methods [5–8] have been applied to pattern recognition problems.

In this paper, we introduce a novel nearest neighbor method. Our method distinguishes itself from earlier implementations of nearest neighbor methods. First, the distance measure is not a simple Euclidean measure, but a weighted one where the weights have some monotonic dependence on internal indices of patterns. Second, we can optimize the weight parameters defining the distance measure in the pattern space by using a training set. Third, we incorporate a regression-like process of reduction of the feature space, which is analogous to the second-layer prediction in neural network methods. We find that these features, especially the second-layer prediction scheme, significantly improve the efficiency of our method.

As a first application of our pattern recognition method, we predict the secondary structure of a given protein. First, we construct the pattern database consisting of about two million patterns. Then, we apply our method to the CB513 set with 513 non-homologous proteins (http://www.compbio.dundee.ac.uk/∼www-jpred/data/) for benchmarking and 29 CASP5 targets

*E-mail: jlee@kias.re.kr; Fax: +82-2-958-3786

(http://predictioncenter.llnl.gov/casp5/) for blind tests, and find that the prediction results are excellent.

II. NEAREST NEIGHBOR METHOD

The basic idea of the nearest neighbor method [3] is as follows. First, one constructs a pattern database from patterns with known classification, $D_M = \{(X_i, Y_i), 1 \leq i \leq M\}$, where X_i is a pattern with class Y_i. Then, the N nearest neighbors closest to a pattern X with unknown classification are identified as

$$\{(X_1, Y_1), \cdots, (X_N, Y_N)\}, \tag{1}$$

by using a distance measure $d(X_i, X_j)$ between X_i and X_j. The class that appears most frequently among the N nearest neighbors,

$$Y = \text{majority}(Y_1, \cdots, Y_N), \tag{2}$$

is predicted for X.

In order to use nearest neighbor methods, it is important to generate patterns with an appropriate distance measure, so that the patterns with identical classifications are clustered in small regions of the pattern space. With such a suitable definition of patterns and a distance measure, nearest neighbor methods can extract more information hidden inside a given pattern by including many patterns with known classification. The reasoning behind this is as follows: It is a well known fact that biological sequences such as amino-acid sequences or DNA sequences are not random, and consequently, the patterns generated from these sequences would occupy only a small portion of the pattern space, compared to those generated from random sequences. Therefore, it is reasonable to imagine that these patterns form islands in the pattern space, where each island consists of patterns of identical classification. In the conventional pattern recognition methods such as neural networks, it is crucial to prepare a non-homologous training set of biological sequences in order to avoid biased outcomes [9]. Inclusion of highly homologous sequences in the training set often leads to worse performance in these methods. This is due to the fact that the neural network is a sort of global classifier in that the whole pattern space is divided by a mapping function with many parameters. The inclusion of many homologous sequences would change the values of parameters; this has a long-range (global) influence on the classifying boundaries that may adversely affect the classification. On the other hand, nearest neighbor methods can be considered as local methods in that only nearby patterns influence the classification of a query pattern. Therefore, pattern recognition based on nearest neighbors does not suffer from the inclusion of highly homologous sequences (Fig. 1).

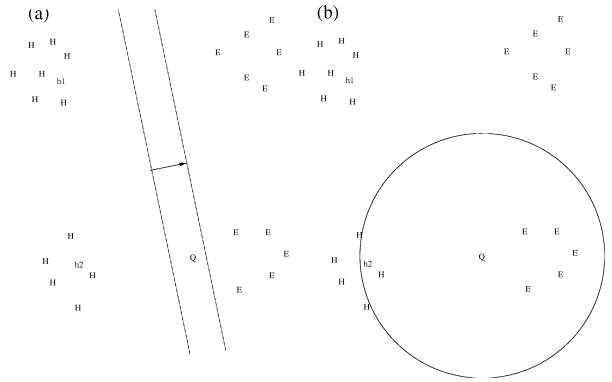

Fig. 1. Schematic figure in the pattern space, comparing the nearest neighbor method with other pattern recognition methods such as neural networks. The letters H and E denote patterns from a non-homologous set of proteins, corresponding to helix and extended secondary structures, respectively. The letter Q denotes a query pattern with an extended conformation. (a) Many pattern recognition methods use these conformations as a training set and construct a boundary between helices and extended conformations. The addition of redundant patterns corresponding to helices, denoted by h1 and h2, may adversely affect the prediction, due to the shift of the boundary as schematically indicated by an arrow in the figure. (b) In the nearest neighbor method, the secondary structure is predicted by enumerating all patterns in the database to determine patterns close in distance to the query pattern, which are enclosed by a circle in the figure. Therefore, the presence of h1 does not affect the prediction result, whereas that of h2 increases the chance that the query pattern is incorrectly predicted as a helix.

1. Protein Secondary Structure and Patterns

Here we consider, as an example of patterns and nearest neighbor methods, the problem of predicting protein secondary structure. Proteins are polymers (chain molecules) built from 20 basic units, called amino acids. The function and biological role of a protein are mainly determined by its three-dimensional geometric structure in its native state. It is well known that native structure is determined solely by the amino-acid sequence information [10]. It is, therefore, a very important task to predict the tertiary structure of a protein, based on the sequence information alone. However, this kind of *ab initio* prediction has not yet been successful to the level of practical applicability.

A less ambitious attempt is to predict, instead of the full 3-D structure, so-called secondary structure elements of local residues (amino acids) that represent local structural information, such as α-helix, β-strand and coil. Reliable prediction of protein secondary structure can serve as an intermediate step toward determining protein tertiary structure. For this reason, much research effort [11–14] has been made for the determination of protein secondary structure. Predicting the secondary structure of a protein from its sequence is a typical pattern recognition problem.

There are many proteins with distinct sequences which

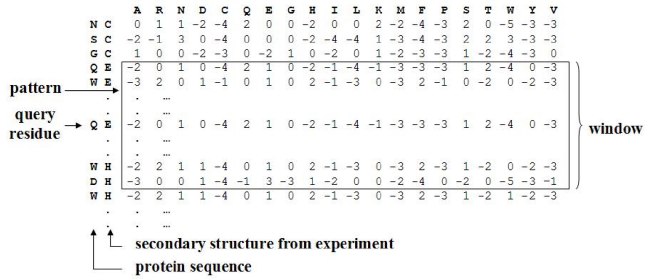

Fig. 2. Definition of the pattern of a residue on the profile by considering a window centered at the residue. For a window size of 15, the pattern is a 15×21 matrix, where 21 stands for 20 amino-acid types plus one indicating vacancies at terminal ends of sequences.

share common secondary structures. Therefore, it is important to find a pattern which represents common features of local environments of protein residues sharing the same secondary structure. For this purpose, one of the most powerful tools available at present is the so-called PSI-BLAST (http://ncbi.nlm.nih.gov/BLAST/) [15]. A rough sketch of steps of PSI-BLAST is as follows. First, for a given sequence, similar sequences are searched from a sequence database using a score function [16]. Thus obtained similar sequences are used to produce substitution frequencies of each residue into 20 different types of amino acids. These tables of substitution frequencies are again used to search the sequence database for recruiting further similar sequences. These newly recruited sequences are now added to the already obtained similar sequences to generate a new table of substitution frequencies. These processes are repeated until no new sequences are found with a preset level of similarity. In this way, we find the final table of substitution frequencies which is the position-specific score matrix (PSSM), also called the profile [17,18]. By performing these kinds of iterative searches, PSI-BLAST is known to find distantly related sequences that are often missed in the initial search.

The profile of a sequence contains information on common features of related sequences. Based on the profile, we define the pattern for each residue by considering seven neighboring residues to the left and to the right of the given residue position, so that the size of the window becomes 15 (Fig. 2). Each pattern contains information on the local environment of a given residue. The concept of this pattern has been used for secondary structure prediction by using neural networks [9,14]. In this work, it is used for the first time in the context of nearest neighbor methods.

The Protein Data Bank (PDB) consists of about 22,000 proteins whose structures have been determined by X-ray crystallography or NMR methods. We construct a pattern database by using the set of 7,777 proteins from the Protein Data Bank (http://www.rcsb.org) after removing identical sequences, considering only the first chains. The structures of these proteins are all determined by X-ray crystallography with resolution better than 3.0 Å. The secondary structure element for a given local residue can be determined by looking into the patterns of hydrogen bonds and geometric features. One of the most widely accepted criteria is the so-called DSSP (Definition of Secondary Structure of Proteins) [19] which automatically generates (from the tertiary structures) an unambiguous and physically meaningful state of secondary structures of proteins. Actually, the original DSSP produces an eight-state classification which, in usual secondary structure predictions, is reduced to three states (H: α-helix, E: β-strand and C: coil). The profiles of 7,777 proteins are generated by PSI-BLAST (with default option $E = 0.001$ and three iterations) for each protein against the NR (nonredundant) sequence database (ftp://ncbi.nlm.nih.gov/blast/db/). The resulting pattern database contains 1,988,085 patterns, which we call the first-layer database.

2. The First-layer Prediction

The distance between two patterns is defined by

$$D_{ij} = \sum_k W_k^0 |P_i^{(k)} - P_j^{(k)}|, \tag{3}$$

where $P_i^{(k)}$ $(k = 1, 2, \ldots, 15 \times 21)$ is the k-th component of the pattern i, and $\{W_k^0\}$ are the weight parameters. Since we expect the pattern components nearer to the center residue to be more important in defining the distance, we use weights $W_k^0 = (8 - |8 - r|)^2$, where r $(r = 1, 2, \ldots, 15)$ is the index labeling the residue position corresponding to the k-th component.

We enumerate all pairwise distances between a query pattern and all the patterns in the database, and select the N nearest patterns. We then simply count the occurrences of H, E and C among these N patterns. The secondary structure of a query pattern can be determined simply by the majority rule that the secondary structure of the most occurrences is chosen for the prediction. We call this procedure the first-layer prediction. The cutoff number N can be chosen suitably by trial and error.

3. The Second-layer Prediction

We introduce the second-layer prediction, which is more efficient than the first-layer prediction. Namely, instead of applying the majority rule from the occurrences of H, E and C, one can construct another kind of pattern based on the first-layer calculation in the following way. The result of the first-layer prediction provides us with a three-state (H, E, C) frequency table for each query residue. These frequency tables provide us with another kind of pattern by considering a window of size 15 on

each residue. We call the resulting pattern the second-layer pattern, which consists of 15×4 elements, where the additional fourth column is used to denote vacancies at the terminal ends of sequences. Therefore, by performing the first-layer calculations for protein residues whose secondary structures are known, we construct the database of the second-layer patterns, which we call the second-layer pattern database. Because of an enormous amount of computational resources required to include all 7,777 protein sequences (1,988,085 residues), we use the CB513 set (84,119 residues) to construct the second-layer database. No two proteins in the CB513 set share more than 25 % sequence identity.

In order to perform the second-layer prediction, we first perform the first-layer prediction for a query residue to obtain the corresponding second-layer pattern. Then, the prediction is performed by comparing the query second-layer pattern with those of the second-layer pattern database. The distance measure is again defined by

$$D_{ij} = \sum_k \tilde{W}_k |S_i^{(k)} - S_j^{(k)}|, \tag{4}$$

where $S_i^{(k)}$ $(k = 1, 2, \ldots, 15 \times 4)$ is the k-th component of the pattern i, and $\tilde{W}_k = (8 - |8 - r|)^2$. Using this distance measure, we select the nearest N patterns for the query residue. Then, the secondary structure of the query residue is predicted following the majority rule by using these nearest N patterns. We use the same value of N that is used in the first-layer calculation. Applying the second-layer procedure improves the Q_3 score (the percentage of correctly predicted residues in sequences of known structure) of the prediction substantially over the one based on the first-layer method alone. One might consider repeating a similar procedure again (this would constitute the third-layer prediction), which we do not carry out in this work.

4. Weight Optimization

For the distance between two first-layer patterns, Eq. (3), we use the initial weights $W_k^0 = (8-|8-r|)^2$. The parameter $\{W_k^0\}$ is not the best one, and it would be desirable to seek better weight parameters. Therefore, we optimize the parameters so that the success rate of our prediction increases for a given set of proteins, called a training set. We choose the CB513 set as the training set. We first perform the first-layer prediction for each residue in this set, by using all the other residues in the CB513 set to generate a pattern database. We select three sets of 100 nearest patterns whose secondary structures are H, E, and C. We calculate the average distances between the patterns in each set and the query pattern, denoted by D_H, D_E, and D_C. We will use the secondary structure corresponding to the least value among D_H, D_E, and D_C as the prediction. The fraction of residues of the proteins in the training set whose secondary structure is correctly predicted by using the initial parameters $\{W_k^0\}$ is 71.0 %.

In order to optimize the parameters, we first define the gaps g_1 and g_2 as follows. Suppose that the secondary structure of a given query residue is a helix. In this case, the gaps are defined as:

$$g_1 = D_H - D_C, \quad g_2 = D_H - D_E. \tag{5}$$

For residues with secondary structures of E or C, the gaps are defined in a similar way. It is obvious that, for a residue whose experimental secondary structure agrees with the prediction, both g_1 and g_2 are negative. Here, we want to change the parameters so that many residues in the training set are predicted correctly. We first select residues whose correct secondary structures differ from the prediction, and call the resulting subset set A. We define $g = \max(g_1, g_2)$, and choose the lowest 10 % (in the value of g) of the residues in set A. We call the resulting subset set B. The residues in set B are considered as the ones whose gaps can be easily converted into negative values by parameter optimization. We then minimize g for all residues in set B, one by one. For each residue, the gaps are linear functions of the weight parameters,

$$g_1 = \sum_k W_k d_1^k, \quad g_2 = \sum_k W_k d_2^k, \tag{6}$$

where the components d_j^k can be easily calculated from the pattern elements. If $g_1 > g_2$, we increase the parameters $\{W_k\}$ by the amount of δW_k:

$$\delta W_k = -\epsilon \ \mathrm{sign}(d_1^k) W_k, \tag{7}$$

where ϵ is a small positive number. Similarly, for $g_1 < g_2$,

$$\delta W_k = -\epsilon \ \mathrm{sign}(d_2^k) W_k. \tag{8}$$

We repeat this procedure 50 times for each residue in set B. When all residues in set B have been used for parameter optimization, one iteration is completed. We start the next iteration by evaluating the gaps of all residues in the training set, selecting the residues with incorrect secondary structure prediction results, selecting the 10 % among them with the smallest gaps and minimizing these gaps. We have performed 300 iterations, and call the resulting parameters $\{W_k^{300}\}$. We have used $\epsilon = 0.2/N_p$, where $N_p = 84,119$ is the number of patterns in the training set. After the parameter optimization, the fraction of residues with negative gaps has increased to Q_3 = 73.1 % from the initial value of Q_3 = 71.0 %.

It should be noted that, as we modify parameters to minimize gaps for a particular residue, the gaps for other residues might increase as a result. For this reason we use a very small value of ϵ, inversely proportional to N_p. Of course, to treat the problem more rigorously, we might consider optimizing gaps for a particular residue while imposing (linear) constraints on the gaps of other residues. This results in an optimization problem where

Table 1. Prediction results on the CB513 set (averaged over 84,199 residues). Q_3^0 and Q_3^{300} denote the Q_3 scores for predictions by using the initial parameters W^0 and the optimized parameters W^{300}, respectively. N is the number of nearest patterns utilized for prediction by the majority rule.

layer	first layer					second layer				
N	100	200	300	400	500	100	200	300	400	500
Q_3^0	74.3	73.1	72.3	71.8	71.6	80.3	77.9	76.5	75.8	75.2
Q_3^{300}	75.4	74.3	73.6	73.2	73.0	80.6	78.3	77.2	76.5	76.1

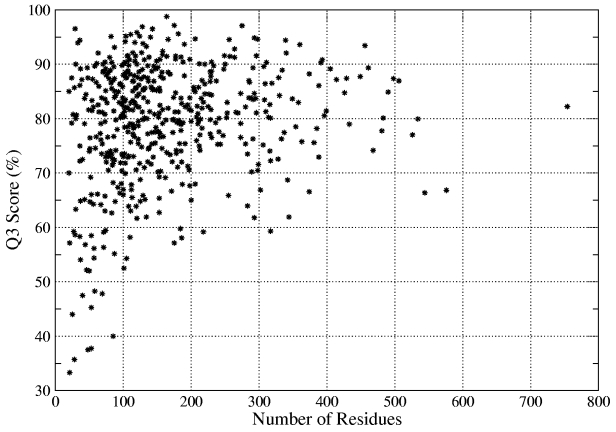

Fig. 3. Q_3 scores as a function of the length of protein sequences (the number of residues) for all 513 proteins in the CB513 set. We have used $N = 100$ and the optimized weight parameters W_k^{300}.

the object function and constraints are all linear functions, called linear programming. We could not pursue this line of investigation, due to the large computer memory requirement. Also, in principle, one might consider a training set containing a more extensive number of proteins than is included in the CB513 set. This would require more extensive computational resources.

III. SECONDARY STRUCTURE PREDICTION

In order to test the performance of our method, we have applied it to secondary structure prediction on the CB513 set. Since most proteins in the CB513 set are included in the pattern database, we paid special attention to the exclusion of the patterns coming from the query sequence in the pattern database, for fair benchmarking. The benchmark results for the CB513 set are shown in Table 1, where $\{W_k^{300}\}$ is used for the first-layer calculation. All second-layer calculations are carried out by using $\{W_k^0\}$.

We observe that the performance of the second-layer prediction is consistently better than that of the first-layer one by about 5 %. It should be noted that, even with the initial parameters $\{W_k^0\}$, the performance is quite excellent. In fact, based on the results of the CB513 set and the CASP5 targets, the efficiency of the parameters $\{W_k^{300}\}$ is only slightly better than that of $\{W_k^0\}$. As a whole, we find that the second-layer prediction with $N = 100$ using the optimized parameters $\{W_k^{300}\}$ in the first-layer calculation gives the best performance of the Q_3 score (80.6 %). Fig. 3 shows the Q_3 scores as a function of the length of protein sequences (the number of residues) for all 513 proteins in the CB513 set. The query patterns of the proteins with the low Q_3 score are generally located rather far from those in the pattern database, implying that these patterns are relatively isolated in the pattern space. That is, the local environments of these residues are quite different from those in the pattern database.

We also worked in CASP5 and applied our method

Table 2. Prediction results on 29 CASP5 targets.

Target	Length	Q_3(Total)	$Q_3(C)$	$Q_3(H)$	$Q_3(E)$
T0129	170	76.47	75.00	77.27	100.00
T0130	100	82.00	65.85	92.31	95.00
T0133	293	77.47	67.86	81.64	50.00
T0137	133	91.73	97.22	75.00	93.51
T0138	135	74.07	61.90	91.67	70.83
T0139	62	62.90	63.64	62.50	100.00
T0141	187	72.19	81.25	70.45	41.94
T0142	280	73.21	70.80	89.36	68.75
T0146	299	63.21	62.15	77.05	52.46
T0147	244	85.66	80.18	96.70	76.19
T0148	162	79.63	75.51	87.69	72.92
T0149	317	79.50	85.16	82.11	69.15
T0150	97	79.38	76.47	75.00	94.74
T0153	134	83.58	80.00	50.00	90.91
T0159	309	79.61	88.50	78.68	65.00
T0160	126	81.75	84.75	58.33	83.64
T0165	318	83.96	88.46	82.11	78.46
T0169	156	75.64	71.88	84.09	72.92
T0170	69	88.41	77.78	95.24	100.00
T0172	295	77.63	75.68	86.23	56.52
T0182	249	85.94	94.39	94.67	62.69
T0183	247	78.14	73.49	83.46	70.27
T0184	240	82.50	82.54	84.97	66.67
T0185	457	81.84	79.80	94.87	66.02
T0186	363	74.66	78.34	84.85	59.81
T0187	417	82.01	75.98	88.89	81.58
T0188	107	83.18	88.64	80.65	78.12
T0189	319	84.33	81.75	90.52	79.22
T0190	111	92.79	90.74	100.00	94.00
Mean (by chain)		79.77			
Mean (by residue)		79.50			

to predict the secondary structure of CASP5 targets in blind fashion. The results are summarized in Table 2. We used the optimized parameters $\{W_k^{300}\}$ with $N = 100$. The average Q_3 score (averaged by the number of chains) for 29 targets is 79.77 %, with a standard deviation of 6.8 %.

IV. CONCLUSION

We have presented a nearest neighbor method for pattern recognition based on a weighted distance measure between patterns generated from profiles. By implementing the second-layer pattern space and profiles, the performance of the prediction is shown to increase significantly. We have also optimized the weight parameters used for the definition of the distance measure, which slightly enhances the performance of the method. We applied this method to the secondary structure prediction of proteins. The performance on the CB513 set and the CASP5 targets is quite impressive, and almost comparable to the state-of-the-art method based on neural network methods [9].

ACKNOWLEDGMENTS

This work was supported by grants No. R01-2003-000-11595-0 (Sung Jong Lee and Jooyoung Lee) and No. R01-2003-000-10999-0 (Julian Lee) from the Basic Research Program of the Korea Science & Engineering Foundation.

REFERENCES

[1] H. C. Andrews, *Introduction to Mathematical Techniques in Pattern Recognition* (Wiley-Interscience, New York, 1972).
[2] H. Grote, Rep. Prog. Phys. **50**, 473 (1987).
[3] S. R. Kulkarni, G. Lugosi and S. S. Venkatesh, IEEE Trans. Inform. Theory **44**, 2178 (1998).
[4] K. Fukunaga, *Introduction to Statistical Pattern Recognition* (Academic Press, New York, 1990).
[5] H. D. Block, Rev. Mod. Phys. **34**, 123 (1962); H. D. Block, Rev. Mod. Phys. **34**, 135 (1962).
[6] T. Shimizu, J. Korean Phys. Soc. **40**, 1072 (2002).
[7] B. Kim, K.-H. Kwon, S.-K. Kwon, J.-M. Park, S.-W. Yoo, K.-S. Park, I.-K. You and B.-W. Kim, J. Korean Phys. Soc. **41**, 433 (2002).
[8] S. Fujiki, M. Nakao and N. M. Fujiki, J. Korean Phys. Soc. **40**, 1091 (2002).
[9] D. T. Jones, J. Mol. Biol. **292**, 192 (1999).
[10] C. B. Anfinsen, Science **181**, 223 (1973).
[11] P. Y. Chou and G. D. Fasman, Biochemistry **13**, 211 (1974).
[12] J. Garnier, D. Osguthorpe and B. Robson, J. Mol. Biol. **120**, 97 (1978).
[13] T. M. Yi and E. S. Lander, J. Mol. Biol. **232**, 1117 (1993).
[14] B. Rost, J. Struc. Biol. **134**, 202 (2001).
[15] S. F. Altschul, T. L. Madden, A. A. Schaffer, J. H. Zhang, Z. Zhang, W. Miller and D. J. Lipman, Nucleic Acids Res. **25**, 3389 (1997).
[16] S. F. Altschul, W. Gish, W. Miller, E. W. Myers and D. J. Lipman, J. Mol. Biol. **215**, 403 (1990).
[17] M. Gribskov, A. D. McLachlan and A. D. Eisenberg, Proc. Natl. Acad. Sci. USA **84**, 4355 (1987).
[18] H. Carrillo and D. Lipman, SIAM J. Appl. Math. **48**, 1073 (1988).
[19] W. Kabsch and C. Sander, Biopolymers **22**, 2577 (1983).