Protein Science (1995), *4*:275–285. Cambridge University Press. Printed in the USA.

Neural networks for secondary structure and structural class predictions

JOHN-MARC CHANDONIA[1] AND MARTIN KARPLUS[1,2]

[1] Biophysics Program, Harvard University, Cambridge, Massachusetts 02138
[2] Department of Chemistry, FAS, Harvard University, Cambridge, Massachusetts 02138

(RECEIVED June 24, 1994; ACCEPTED September 16, 1994)

Abstract

A pair of neural network-based algorithms is presented for predicting the tertiary structural class and the secondary structure of proteins. Each algorithm realizes improvements in accuracy based on information provided by the other. Structural class prediction of proteins nonhomologous to any in the training set is improved significantly, from 62.3% to 73.9%, and secondary structure prediction accuracy improves slightly, from 62.26% to 62.64%. A number of aspects of neural network optimization and testing are examined. They include network overtraining and an output filter based on a rolling average. Secondary structure prediction results vary greatly depending on the particular proteins chosen for the training and test sets; consequently, an appropriate measure of accuracy reflects the more unbiased approach of "jackknife" cross-validation (testing each protein in the database individually).

Keywords: neural networks; secondary structure prediction; structural class prediction

Predicting the structure of a protein from the primary amino acid sequence is one of the fundamental problems in computational biology. Much effort has been directed at the prediction of secondary structure. Recent applications of a variety of techniques, such as neural networks, Bayesian statistics, and other pattern recognition methods have obtained 3-state prediction accuracies (helix, sheet, other) of 62.7–64.4% (Qian & Sejnowski, 1988; Holley & Karplus, 1989, 1991; Stolorz et al., 1991). This appears to be near the limit for unbiased secondary structure prediction of a single protein sequence. By creating profiles of aligned, homologous sequences, and training and testing neural networks on these rather than on individual proteins, Rost and Sander have obtained substantial improvements, with an average 3-state prediction accuracy of 72.5% on sequences nonhomologous to any in the training set (Rost & Sander, 1994). Few modifications to the underlying neural network, relative to that used in the single sequence studies, were made. Although prediction accuracy may improve with the addition of more well-resolved protein structures (Rooman & Wodak, 1988), much of the inaccuracy in current secondary structure prediction methods is believed to be due to the lack of consideration of long-range interactions that arise from the (unknown) tertiary structure. This is a consequence of the fact that many sequences have alternative secondary structural possibilities (Kabsch & Sander, 1984; Argos, 1987; Holley & Karplus, 1991).

It has been found that basic information on protein tertiary structure such as the folding class (i.e., all-α, all-β, . . . , as defined by Levitt & Chothia [1976]) can be helpful in improving the accuracy of secondary structure prediction (Taylor & Thornton, 1984; Kneller et al., 1990; Presnell et al., 1992). Kneller et al. found that prediction accuracy on proteins in the all-α class was improved by 16% (from 63% to 79%) by using a neural network trained on similar proteins. Accuracy on proteins in the all-β class improved by 6%, from 63% to 69%. Accuracy on α/β proteins did not improve and other classes were not examined. The observed increase in accuracy was due partially to the fact that proteins used in the training set were more similar in tertiary structure to the predicted proteins than in a set including all protein classes, as determined by their secondary structure content and range of possible folds. Also, some of the increased accuracy was due to the reduction of the secondary structure prediction problem from the standard 3-state prediction (helix, sheet, and coil) to one of 2 states (coil and either helix or sheet for the all-α and all-β proteins, respectively). In the rare cases in which β-sheets appeared in proteins of the all-α class, they were never predicted by the network. Because information on the structural class may be obtained experimentally (e.g., by spectroscopic methods such as CD) with significantly greater ease than the determination of a high-resolution 3D structure, class-based secondary structure prediction could be useful in

Reprint requests to: Martin Karplus, Department of Chemistry, 12 Oxford Street, Cambridge, Massachusetts 02138; e-mail: marci@tammy.harvard.edu.

practice. However, a purely computational approach combining class and secondary structure prediction would be much more desirable because it could be applied immediately to any available sequence.

Efforts at de novo folding class prediction have met with limited success. Although overall helix and sheet content can be predicted with less than 10% error (Muskal & Kim, 1992; Rost & Sander, 1993b), this margin of error is too large to identify proteins in the all-α and all-β classes. One difficulty in identifying protein classes is that there are no clear-cut boundaries in secondary structure content between proteins from the 4 classes (Rost & Sander, 1994). Rost and Sander (1993b) found that only 58% of all-α proteins could be identified, with several non-α proteins misclassified as all-α. They found a 3% increase in secondary structure prediction accuracy when these proteins were tested using a network trained on other all-α proteins. Unfortunately, the decrease in accuracy on the proteins misclassified as all-α outweighed this gain. Only 50% of proteins in the all-β class could be correctly identified, with several proteins misclassified as all-β. Statistical algorithms have been developed which claim 70% accuracy at assigning a protein to one of 5 classes, or 83% accuracy with 4 classes (Klein & DeLisi 1986; Chou, 1989; Zhang & Chou, 1992). However, some of the sets of proteins on which these algorithms were tested contained high levels of sequence homology (more than 90% identity in some cases) with each other and with the proteins used in determining the numeric parameters of the algorithm. An unbiased test in which these algorithms are applied to proteins without significant sequence homology has not been done.

Neural networks have yielded promising results in identifying specific tertiary folds with no experimental information besides the amino acid content and length (Dubchak et al., 1993). An accuracy of 87% was achieved at distinguishing proteins of 4 specific folds: 4-helix bundles, parallel $(\alpha/\beta)_8$ barrels, nucleotide binding fold, and immunoglobulins. Although these results are of interest, the folds that were tested are very different from each other in helix and sheet content, amino acid composition, and size; proteins with the same fold show little variation in these parameters. Thus, this algorithm may be insufficient for distinguishing proteins of more similar folds, without introducing additional parameters such as those considered here.

In this paper, we show that information obtained from a secondary structure prediction algorithm can be used to improve the accuracy of a neural network for the de novo prediction of the folding class. Furthermore, the results of the folding class prediction contain some tertiary structural information that is useful for improving the results of secondary structure prediction. This iterative approach yields better results than either prediction applied independently. The approach can be summarized as follows: (1) A secondary structure prediction for a protein is obtained using standard neural network techniques with the amino acid sequence as input. (2) Information from this prediction and other data obtained from the sequence (such as the length and the amino acid content) are provided to another neural network, which predicts the structural class of the protein. (3) The structure class prediction is used in conjunction with the sequence information by a third network, which produces a slightly more accurate secondary structure prediction. This procedure can be repeated. In this paper, we apply the integrated approach to a commonly used set of proteins (Kabsch & Sander, 1983b) and compare it with independent structural class and secondary structure prediction methods. Some cautions concerning this approach, and use of neural networks in general, are presented and discussed.

Methods

Neural networks

All neural networks used in this model are standard feed-forward networks consisting of 2 or 3 layers of units (Rumelhart et al., 1986; Holley & Karplus, 1991). They are fully connected from one layer to the next. The first and last layers are referred to as the input and output layers, respectively. The middle layer, if present, is referred to as the hidden layer because its inputs and outputs connect only to other network units, rather than representing physical data (i.e., an amino acid sequence or secondary structure).

Each unit in the neural network accepts a number of inputs from units in the previous layer, or from external data in the case of the input layer. Each input is multiplied by a weight W_{ij}, which represents the strength of the connection between 2 units i and j, and the total is offset by the bias b_i of the unit:

$$input_i = \sum_j W_{ij} + b_i. \quad (1)$$

The unit processes this input using a continuous, nonlinear "activation function" that switches from near 0 to near 1 over a fairly narrow threshold. The following function is used here:

$$output_i = \frac{1}{1 + e^{-input_i}}. \quad (2)$$

The network is made up of units that act as a set of nonlinear functions between the initial input and the final output. The independent variables in these functions are the biases of each individual unit, b_i, and the weights between every pair of units in adjacent layers, W_{ij}. These variables are initialized with small, random numbers; they converge to useful values based on input data through an iterative process that is referred to as *training* the network.

In practice, several modifications to this model were introduced to improve the speed of the algorithm presented by Holley and Karplus (1989). First, units in the input layer simply pass their input through as their output, rather than using the formula given in Equation 2. This is possible because our activation function is one to one: each possible output corresponds to a unique input. Therefore, the numerical values of the inputs are somewhat arbitrary as long as the encoding scheme is preserved between training and prediction (Rumelhart et al., 1986). This modification improves the speed of the algorithm because multiple calculations of the nonlinear activation function (Equation 2) can be avoided when training (and using) the network. Also, the bias b_i is implemented as a weight W_{0i} from an additional unit in the previous layer that is always turned on (output = 1.0). For units not in the input layer, this sum of weights from the previous layer is mathematically equivalent to Equation 1, and produces a slight simplification in the code. Because

the input layer is the first layer of the network, units in the input layer have their bias set to zero. Because the input layer can consist of many units (as in the networks presented here), a large number of variables are eliminated from the training procedure, resulting in faster convergence of the remaining weights and biases.

Networks are trained on a set of data for which the desired output is known; this is referred to as the *training set*. The method used is back-propagation, a well-characterized algorithm for adjusting the weights and biases (Rumelhart et al., 1986). For sparse data sets (when compared to the total number of independent variables in the network) the network may "memorize" features of the training set rather than learning general features applicable to data outside the training set (Rumelhart et al., 1986). We have measured the degree to which this takes place for our data and have taken steps to eliminate the effects of overtraining.

After training, the network can be exposed to new data for which the desired output is not known to the network; this is known as the *test set*. To ensure unbiased testing, data in the test set should be dissimilar to data already presented to the network in the training phase. Qian and Sejnowski (1988) have shown that the secondary structure of proteins homologous to those in the training set is generally predicted with higher accuracy than that of unrelated proteins. For our networks, in which protein features were used as input, no proteins in the training set showed significant sequence homology to proteins in the test set.

Data set

The proteins used in this study were a set of 62 globular proteins (69 chains) used in several previous studies of secondary structure prediction methods (Kabsch & Sander, 1983a; Holley & Karplus, 1991). All proteins in the database were refined to 3.0 Å or better. This data set includes examples of all globular folds for which a well-resolved structure was published prior to 1983. No pair of protein chains in the data set contains more than 47% sequence identity.

The program DSSP (Kabsch & Sander, 1983b) was used to classify the secondary structure of all residues in the database. All secondary structure types besides α-helix (H) and extended β-sheet (E) were collapsed into the "coil" category. The complete database contains a total of 10,767 residues with a composition of 26% helix, 20% sheet, and 54% coil; 3_{10}-helices were treated as coil.

Three different methods for partitioning this data set into training and test sets were examined. For comparison with earlier results, most tests were done with the 48-protein training set and 14-protein test sets first used by Kabsch and Sander (1983a). We also divided the database randomly into 10 sets of 48 training proteins and 14 test proteins. These 10 sets were tested on several network topologies to determine the average results and the degree of variation that can occur. Because we found that results vary with the particular proteins chosen for training and test sets, final testing was done using jackknife cross-validation. Networks were trained on training sets produced by removing a single protein from the database. Each network was then tested on the excluded protein and the results were combined for evaluation of overall prediction accuracy.

Secondary structure prediction networks

The network uses a "sliding window" approach to iteratively predict the secondary structure of each residue in the protein. At a given time, the network is presented with 15–27 (the *window width*) sequential residues of the protein. When training or testing the network, this input window is centered on each of the residues in the protein in turn, and produces a secondary structure prediction for that residue. An overview of this network is shown in Figure 1 and each layer is discussed below.

For each residue in the input window, the residue type is encoded and presented to the network in 21 separate units of the input layer. Twenty of the units represent a single amino acid residue and are turned on (input = 1) or off (input = 0) depending on whether that residue appears at the particular position in the window. The 21st unit is turned on when no amino acid appears at that position; this occurs when the window overlaps the ends of the sequence. Thus, 21 units total are used in the input layer for each amino acid in the input window.

The output layer of the networks consisted of 2 units, h and e, whose outputs correspond to helix and sheet prediction, respectively. During training, residues in an α-helix were trained using desired outputs of $h = 0.95$, $e = 0.05$. These values represent extremes of the output function given in Equation 2, which is sigmoidal and approaches 0 and 1 in the limits of infinitely low and high input. Training with desired outputs set to 0 or 1 can result in infinite weights, so values close to the upper and lower limits of the output function were chosen. These values yield good results and require a reasonable time for network training. Residues in a β-sheet were trained with the desired outputs $h = 0.05$, $e = 0.95$. All other residues ("coil" residues) were trained with the desired outputs $h = e = 0.05$. In making a prediction for a residue of unknown secondary structure, the outputs h and e are compared to a cutoff; if neither value was greater than the cutoff, coil is predicted as the secondary structure. If either value is greater than the cutoff, the corresponding secondary structure is predicted; if both are greater, the secondary structure corresponding to the higher of the 2 values is predicted. The cutoff was experimentally determined for each training set; the cutoff value that produced the highest accuracy (as measured by the sum of Matthews correlation coefficients, described below) on proteins in the training set was used. Previous studies have shown a correlation between the magnitude of the network outputs and the accuracy of the prediction. For a similar network, the accuracy of the 31% "strongest" predictions (highest network outputs) were found to be 79% accurate, as opposed to 63% for all outputs (Holley & Karplus, 1991).

Hidden layers of several sizes were tested to determine which produced the most accurate results. A single hidden layer containing 1–20 units was used in each trial. Because several network topologies were tested, we developed a shorthand notation for describing it. The first number in the notation is the width of the input window, in residues (the number of units in the input layer is 21 times as large). If a hidden layer was used, the second number is the number of units in the hidden layer; if not present, the notation contains only 2 numbers corresponding to the size of the input and output layers (i.e., 19×2). The final number is the number of units in the output layer. Thus, a $19 \times 2 \times 2$ network corresponds to a window of 19 residues, with a hidden layer of 2 units, and produces 2 outputs H and E. This network contains a total of 399 input units, 2 units in the hidden

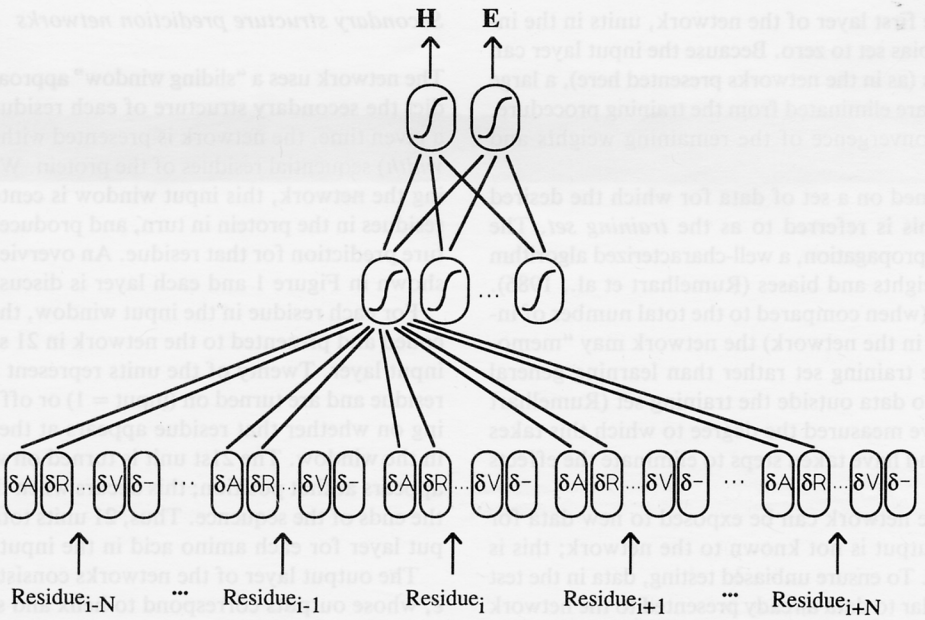

Fig. 1. Secondary structure prediction network. Units in the network are represented by ellipses, connections between units by solid lines. In the input layer, shown at the bottom of the figure, clusters of 21 units are used to input the type of each residue in a continuous stretch of sequence surrounding a given residue, *i*, for which the secondary structure is being predicted. Depending on the identity of the residue, 1 of the 21 units in each cluster is turned on (input of 1); the rest are off (input of 0). These units are labeled *δres*, and are turned on when the residue of type "*res*" occurs at the position. The 21st unit in each cluster is turned on if no residue is present at the position, as occurs when the input window overlaps the ends of the protein. This unit is labeled *δ*−. All input units are connected to every unit in the hidden layer, each of which is connected to both output units, *H* and *E*. Units in the hidden and output layers are labeled with a sigmoidal curve to indicate the sigmoidal relationship betwen the input and the output (Equation 2). Most units and connections are not shown for clarity.

layer, 2 units in the output layer, and 803 connections between units because the network is fully connected between adjacent layers. There are a total of 807 independent variables to optimize, i.e., the weight of each connection and the bias for each unit except those in the input layer.

After all predictions are made, short stretches of helix (<4 residues) and sheet (<2 residues) are filtered to coil; these are the cutoffs used by DSSP for short segments of secondary structure. In addition to this removal of short segments of secondary structure, an additional filter was run on the network outputs prior to their interpretation as secondary structure. In each step of this process, known as *smoothing*, *h* and *e* values from each position were averaged with the *h* and *e* values from residues at adjacent positions:

$$h_i = \frac{h_{i-1} + h_i + h_{i+1}}{3} \tag{3}$$

$$e_i = \frac{e_{i-1} + e_i + e_{i+1}}{3}. \tag{4}$$

Values at the ends of the sequence were averaged only with the single adjacent residue. This process (a step) can be repeated once the entire sequence is processed. Over several steps of smoothing, structural features begin to blur over a larger region of the sequence; this is expected because the equations are similar to those governing 1-dimensional diffusion. This procedure is used to eliminate sharp transitions in the network outputs over short stretches of sequence.

Measurements of accuracy

The most commonly reported measure of secondary structure prediction accuracy is the success rate, or Q_3. This is the overall percentage of correctly predicted residues of all 3 types, i.e.,

$$Q_3 = \frac{R_{helix} + R_{sheet} + R_{coil}}{N}. \tag{5}$$

Here, R_{str} is the number of correctly predicted residues of type *str*, and N is the total number of residues. Although the Q_3 score provides a quick measure of the accuracy of the algorithm, it does not account for differing success rates on different types of secondary structure. We therefore also calculated the correlation coefficients (Matthews, 1975) for prediction of helix (C_H), sheet (C_E), and coil (C_C).

$$C_H = \frac{(p_H n_H) - (u_H o_H)}{\sqrt{(n_H + u_H)(n_H + o_H)(p_H + u_H)(p_H + o_H)}}. \tag{6}$$

In this calculation, p_H is the number of correctly predicted helical residues, n_H is the number of residues that are correctly identified as something other than helix, o_H is the number of nonhelical residues that are predicted as helix, and u_H is the

number of helical residues that are missed by the algorithm. A corresponding calculation is done for C_E and C_C.

Structural class prediction

Quantitative definitions for a 4-class system have been proposed by Kneller et al. (1990) based on examination of typical proteins from the Levitt and Chothia (1976) classes. "All-α" proteins must have at least 75 residues (the size of the Ca-binding protein 3ICB), contain at least 30% α-helix, and contain at least 85% α-helix in regions of well-defined secondary structure. "All-β" proteins must be at least 99 residues long (the size of plastocyanin, 3PCY), with less than 10% helical residues. "α/β" Proteins must be at least 138 residues long (the size of flavodoxin, 4FXN), contain at least 15% α-helix and 5% β-sheet, and have approximate alternation of α and β structure (we quantified this as meaning at least 2 alternations between helix and sheet). Proteins not fitting one of these descriptions are classified "other." The data set contained 14 chains in the all-α class, 15 all-β chains, 16 α/β chains, and 24 other chains. All-α proteins contained an average of 55% helix, 2% sheet, and 43% coil. All-β proteins averaged 4% helix, 36% sheet, and 60% coil. Proteins in the α/β class averaged 30% helix, 17% sheet, and 53% coil. Other proteins averaged 21% helix, 18% sheet, and 61% coil and were too small or contained insufficient helix and sheet to fit any of the other classes.

For predictions of the structural class, the sliding window method employed by secondary structure networks is inadequate because the network must view the entire protein at one time. One way of presenting global information is to provide the network with the amino acid composition of the protein and the sequence length. Dubchak and colleagues (1993) have shown that this information alone is sufficient to train a network to distinguish among several specific tertiary folds that vary significantly in secondary structure content, amino acid composition, and size, as described above.

An overview of the class prediction network is shown in Figure 2. In addition to providing the network with information on sequence length and amino acid composition, data produced by the secondary structure prediction network are also given. Because the class definitions depend on secondary structure, this information is expected to provide a good first-order approximation of the structural class. Along with the original 21 inputs for length and amino acid composition, 2 inputs were provided for the percentage of helix and sheet predicted by the secondary structure network. Two more inputs listed the percentage of "strong" predictions of helix and sheet from the same network (defined as a raw network *h* or *e* output greater than 0.6); the accuracy of these predictions is expected to be higher, although "strong" predictions are infrequent. A final input indicated the expected number of alternations between helix and sheet as one traverses the primary sequence of the protein, also

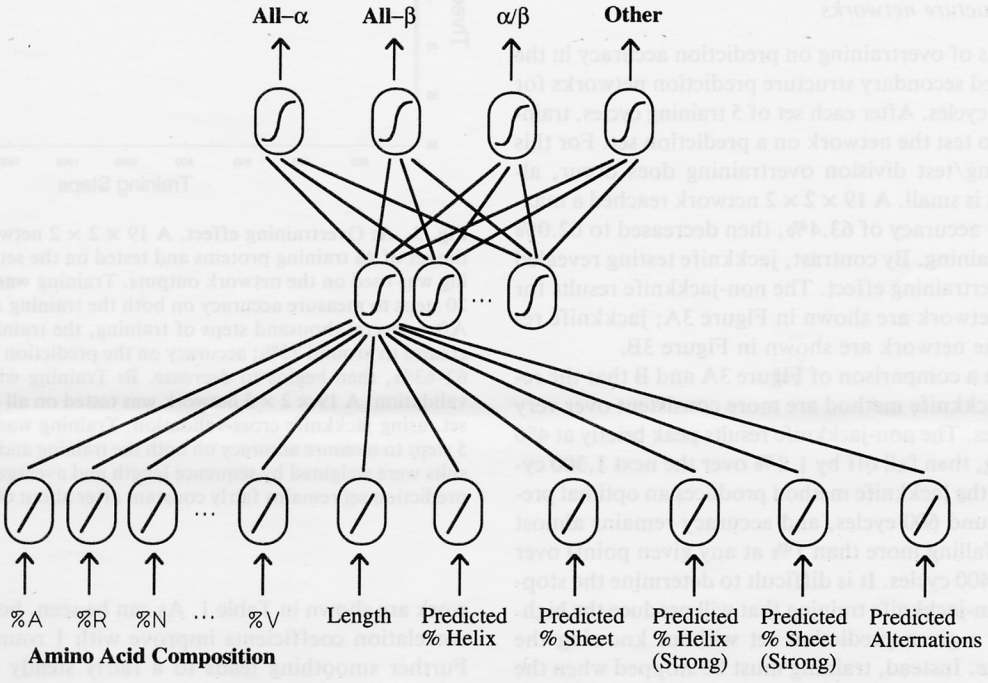

Fig. 2. Class prediction network. Units in the network are represented by ellipses; connections between units by solid lines. Units in the input layer are labeled with linear curves, indicating that their output is equal to their input. Units in the hidden and output layers are labeled with sigmoidal curves, to indicate the sigmoidal relationship between the input and the output (Equation 2). The input layer contains 20 units for describing the amino acid composition of a protein (labeled *%res*), 1 unit for the sequence length, and 5 units containing characteristics of the protein predicted by the secondary structure network. These units indicate the predicted percent helix and sheet, the percentage of strong helix and sheet predictions (which are more accurate), and the predicted number of alternations between helix and sheet. All input units are connected to every unit in the hidden layer, each of which is connected to all output units. In the 4-output network, 1 output is used for each of the defined structural classes: All-α, all-β, α/β, and other. In the single-output version of this network, which predicts whether a protein belongs to a single class, the output layer contains 1 output unit representing the likelihood that a protein belongs to the given class.

as predicted by the secondary structure network. Several of these alternations are usually found in proteins from the α/β class. Results obtained using the full-class prediction network described above are compared to those obtained using a network without any predicted information on secondary structure, i.e., using only the first 21 inputs shown in Figure 2.

Two types of output strategies were tested. A 4-output network with 1 output for each class (as defined by Kneller) directly predicts the class of a protein. The output is interpreted by predicting the class corresponding to the highest of the 4 outputs. In addition, 4 separate single-output networks were trained to specialize in identifying proteins from 1 of the 4 structural classes. These networks were identical in topology to the 4-output network shown in Figure 2, except that each contained only 1 unit in the output layer, indicating the tendency of the tested protein toward each particular class. As in secondary structure prediction, the output unit was compared to a cutoff in deciding whether a given class was predicted. This cutoff was optimized separately in each trial to achieve the highest accuracy on proteins in the training set. The latter method of making independent predictions has the disadvantage that several or none of the 4 classes may be predicted. However, it might allow the network to specialize and produce more accurate predictions of the corresponding class.

Results

Some aspects of the implementation of secondary structure networks

To test the effects of overtraining on prediction accuracy in the test set, we trained secondary structure prediction networks for several thousand cycles. After each set of 5 training cycles, training was paused to test the network on a prediction set. For this particular training/test division overtraining does occur, although the effect is small. A $19 \times 2 \times 2$ network reached a maximum prediction accuracy of 63.4%, then decreased to 62.0% with increased training. By contrast, jackknife testing revealed no significant overtraining effect. The non-jackknife results for the $19 \times 2 \times 2$ network are shown in Figure 3A; jackknife results for the same network are shown in Figure 3B.

It is clear from a comparison of Figure 3A and B that the results using the jackknife method are more consistent over very long training times. The non-jackknife results peak briefly at 480 cycles of training, then fall off by 1.8% over the next 1,300 cycles. In contrast, the jackknife method produces an optimal prediction after around 600 cycles, and accuracy remains almost constant (never falling more than 1% at any given point) over the remaining 1,400 cycles. It is difficult to determine the stopping point for non-jackknife training that will produce the highest accuracy on a given prediction set without knowing the results in advance. Instead, training must be stopped when the decrease in error between several subsequent training steps becomes sufficiently small (we used $\Delta E = 0.06$, which occurs after about 500 steps).

Smoothing

We found that 1 cycle of smoothing decreases training set accuracy while improving prediction results slightly for all networks tested. However, prediction accuracy decreases if more than 1 cycle of smoothing is used. Results for the $19 \times 2 \times 2$ network are shown in Table 1. As can be seen, both the Q_3 and the correlation coefficients improve with 1 round of smoothing. Further smoothing leads to a fairly steady decrease in both measures.

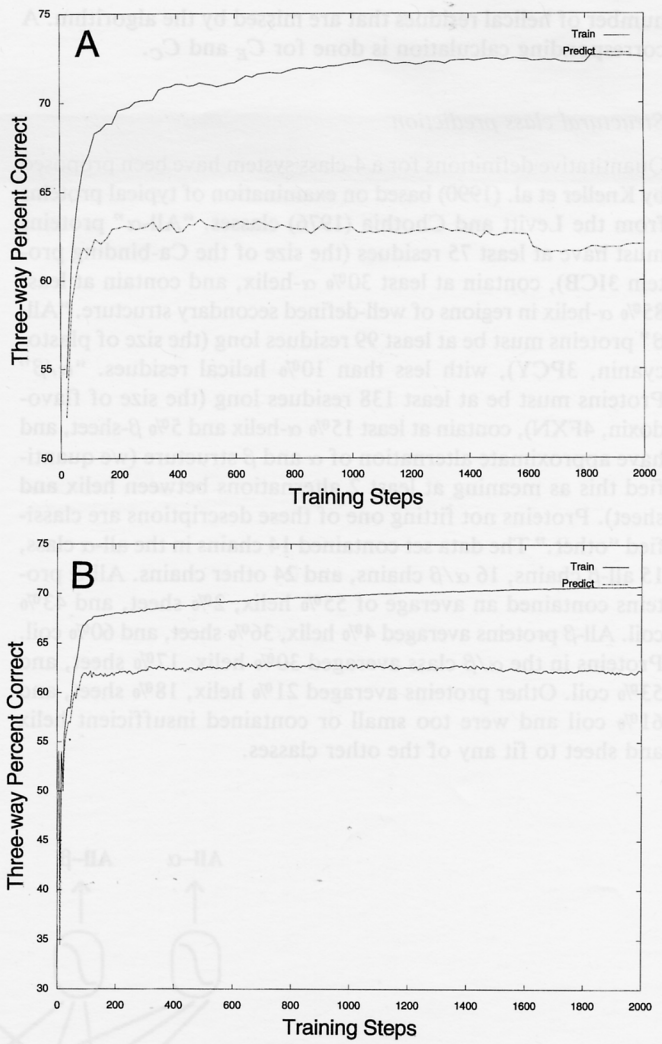

Fig. 3. A: Overtraining effect. A $19 \times 2 \times 2$ network was trained on the set of 48 training proteins and tested on the set of 14. No smoothing was used on the network outputs. Training was paused after every 20 steps to measure accuracy on both the training and prediction sets. After several thousand steps of training, the training set accuracy increases to around 75%; accuracy on the prediction set peaks at around 62–63%, then begins to decrease. **B:** Training with jackknife cross-validation. A $19 \times 2 \times 2$ network was tested on all proteins in the data set, using jackknife cross-validation. Training was paused after every 5 steps to measure accuracy on both the training and prediction sets. Results were weighted by sequence length and averaged. Accuracy on the prediction set remains fairly constant after about 600 steps of training.

Effects of topology

Window widths from 15 to 27 were tested with and without a hidden layer; 19 was found to be optimal. At a window width of 19, seven hidden layer sizes were tested; the network topology with the best prediction success is $19 \times 2 \times 2$ with 1 round of smoothing. As before, these results were compiled on the Holley and Karplus (1989) training and test sets for ease in com-

Table 1. *Secondary structure prediction accuracy as a function of the number of smoothing cycles used*[a]

Smoothing cycles	Prediction accuracy (Q_3)	C_H	C_E	C_C
0	61.9	36	32	35
1	62.8	38	34	38
2	62.2	37	34	37
3	62.4	37	34	37
4	61.7	36	32	36
5	61.9	37	33	36

[a] A neural network was trained on the 48-protein training set and accuracy was measured using the 14-protein test set. Correlation coefficients are multiplied by 100.

parison. Results for all networks tested are shown in Table 2, along with results reported by Holley and Karplus (1989).

Dependence on training and test sets

There was little variation in the results when calculations using the same training and test sets were repeated. This implies that convergence to a set of nearly optimum parameters was obtained in each case. However, results vary widely depending on the choice of training and test sets. Several network topologies were tested on the 10 randomly partitioned training and test sets described in the Methods section. The results are shown in Table 3.

As before, the best results were obtained using the 19 × 2 × 2 network with 1 round of smoothing. However, it is evident that there is a large variation in the results, both for the prediction accuracy and the correlation coefficients. This suggests that considerable care has to be used in evaluating the results of a single partition test. In this study, further evaluations of secondary structure prediction accuracy on the 62-protein database were done using the jackknife method of cross-validation.

Table 2. *Secondary structure prediction accuracy for several different network topologies*[a]

Network topology	Prediction accuracy (Q_3)	C_H	C_E	C_C
15 × 2 × 2	61.8	35	30	34
17 × 2 × 2	61.8	36	32	35
19 × 2 × 2	62.4	37	32	35
21 × 2 × 2	62.3	38	30	36
23 × 2 × 2	63.0	37	32	38
25 × 2 × 2	61.7	36	30	35
27 × 2 × 2	61.7	36	28	34
15 × 2 × 2 (s)[b]	62.9	36	33	41
17 × 2 × 2 (s)	62.5	37	34	37
19 × 2 × 2 (s)	63.6	39	33	39
21 × 2 × 2 (s)	62.6	37	32	36
23 × 2 × 2 (s)	62.6	38	31	37
25 × 2 × 2 (s)	61.8	36	29	35
27 × 2 × 2 (s)	61.5	38	28	35
19 × 1 × 2 (s)	59.5	37	0	28
19 × 3 × 2 (s)	62.8	38	32	37
19 × 4 × 2 (s)	62.6	38	31	36
19 × 5 × 2 (s)	62.8	38	30	36
19 × 6 × 2 (s)	62.3	38	31	36
19 × 10 × 2 (s)	62.6	38	29	36
19 × 20 × 2 (s)	62.2	37	26	35
H & K[c]	63.2	41	32	36

[a] Neural networks were trained on the 48-protein training set and accuracy was measured using the 14-protein test set. Correlation coefficients are multiplied by 100.
[b] (s), One round of smoothing.
[c] H & K, Holley and Karplus (1989).

Structural class prediction

The 4-output and single-output networks were tested on the database using jackknife cross-validation. Networks were first trained and tested without using any predicted information on secondary structure, i.e., using the 21-input topology described in the Methods section, under "Structural class prediction." Training was stopped when the decrease in error was sufficiently small ($\Delta E = 0.01$/step, or about 350 training steps). The results obtained from the 2 types of networks are reported in Tables 4 and 5.

The single-output networks are able to correctly identify 43% of the all-α proteins, 53% of the all-β proteins, 69% of α/β proteins, and 66% of other proteins. Four-output networks performed better on all but 1 class, identifying 57% of the all-α proteins, 60% of the all-β proteins, 63% of the α/β proteins, and 71% of other proteins. Compared with results obtained by classifying proteins directly using secondary structure predictions (Rost & Sander, 1993b), the 4-output network was comparable in classifying all-α proteins (57% versus 58%) and slightly better at classifying proteins in the all-β class (60% versus 50%).

Table 3. *Secondary structure prediction on 10 randomly chosen sets of training and test proteins*[a]

Network topology	Prediction accuracy (Q_3)		C_H	C_E	C_C
17 × 2 × 2	Worst:	58.9	30	33	30
	Best:	65.0	46	39	40
	Average:	61.9	36	32	34
17 × 2 × 2 (s)[b]	Worst:	59.8	35	32	32
	Best:	65.8	48	38	41
	Average:	62.3	37	31	35
19 × 2 × 2	Worst:	58.4	23	29	28
	Best:	65.1	48	34	41
	Average:	61.5	36	31	34
19 × 2 × 2 (s)	Worst:	58.5	24	29	28
	Best:	66.7	51	39	44
	Average:	62.4	38	32	35

[a] Neural networks were trained on randomly chosen sets of 48 proteins from the database and tested on the remaining 14 proteins. The best, worst, and average results for the 10 trials are shown. Correlation coefficients are multiplied by 100.
[b] (s), One round of smoothing.

Table 4. *Structural class prediction using single-output networks without information on secondary structure*[a]

Network	Prediction	Actual class			
		All-α	All-β	α/β	Other
All-α	All-α (17)	6	2	3	6
	Not all-α (52)	8	13	13	18
All-β	All-β (19)	0	8	5	6
	Not all-β (50)	14	7	11	18
α/β	α/β (18)	2	4	11	1
	Not α/β (51)	12	11	5	23
Other	Other (23)	4	3	0	16
	Not other (46)	10	12	16	8

[a] Testing was done using jackknife cross-validation.

Table 6. *Structural class prediction using single-output networks and predicted information on secondary structure*[a]

Network	Prediction	Actual class			
		All-α	All-β	α/β	Other
All-α	All-α (13)	8	0	1	4
	Not all-α (56)	6	15	15	20
All-β	All-β (13)	0	9	2	2
	Not all-β (56)	14	6	14	22
α/β	α/β (14)	1	1	12	0
	Not α/β (55)	13	14	4	24
Other	Other (23)	3	2	0	18
	Not other (46)	11	13	16	6

[a] Testing was done using jackknife cross-validation.

Although this network is similar to that used by Dubchak et al. (1993) to identify 4 specific protein folds, it is clearly more difficult for the network to learn to identify general structural classes; only 62% were correctly identified, compared to 87% of the proteins in the previous work.

To apply the full class prediction network shown in Figure 2, the secondary structure of each protein was predicted using the $19 \times 2 \times 2$ network, with 1 round of smoothing. Class prediction was then done for each protein, using the jackknife procedure of cross-validation. The jackknife procedure was also used to obtain predicted secondary structures for each of the proteins tested by the class prediction network, to prevent any known information on secondary structure content from being used in the test class prediction. However, accurate (rather than predicted) information on secondary structure content was used for proteins in the training sets. This produces more accurate results on both the training and test sets (results not shown). The results obtained using single-output and 4-output networks are shown in Tables 6 and 7, respectively. The overall training and prediction set accuracy versus training time is shown in Figure 4.

It is clear from the tables that overall accuracy is quite good, and much better than without the predicted secondary structure input. The 4-output network correctly identifies 64% of all-α proteins, 73% of all-β proteins, 81% of α/β proteins, 75% of other proteins. As before, the single-output networks were slightly less successful; they correctly identified 57% of all-α proteins, 60% of all-β proteins, and 75% of α/β and other proteins. The addition of a hidden layer to either type of network did not improve the accuracy. Although all-α proteins are the most difficult class for the network to identify, accuracy is slightly higher than a previous result of 58% obtained by directly determining the protein class using highly accurate (70%) secondary structure predictions (Rost & Sander, 1993b).

An important result is that no protein in the all-α class was misclassified as all-β and no protein in the all-β class was classified as all-α (Tables 6, 7). Also, there were no misclassifications between the α/β and "other" classes. The result demonstrates that the class prediction networks can always eliminate one or more classes with accuracy approaching 100%. This can be done by predicting the class of a protein using the 4-output network and all four of the single-output networks, and then eliminating one or more classes based on the predictions. If the protein is predicted as all-α by either the 4-output network or single-output all-α network, all-β is eliminated as a potential class, and vice versa. If a protein is predicted to be α/β by any network, "other" is eliminated, and vice versa. Using both the single-output and 4-output networks for elimination is slightly better than using either alone because the predictions are independent and can sometimes lead to the elimination of more than 1 potential class.

Table 5. *Structural class prediction using a 4-output network without information on secondary structure*[a]

Prediction	Proteins	Actual class			
		All-α	All-β	α/β	Other
All-α	15	7	1	3	4
All-β	15	1	9	2	3
α/β	16	3	3	10	0
Other	23	3	2	1	17

Total: 43/69 (62.32%) predicted correctly

[a] Testing was done using jackknife cross-validation.

Table 7. *Structural class prediction using a 4-output network and predicted information on secondary structure*[a]

Prediction	Proteins	Actual class			
		All-α	All-β	α/β	Other
All-α	12	9	0	1	2
All-β	17	0	11	2	4
α/β	17	2	2	13	0
Other	23	3	2	0	18

Total: 51/69 (73.91%) predicted correctly

[a] Testing was done using jackknife cross-validation.

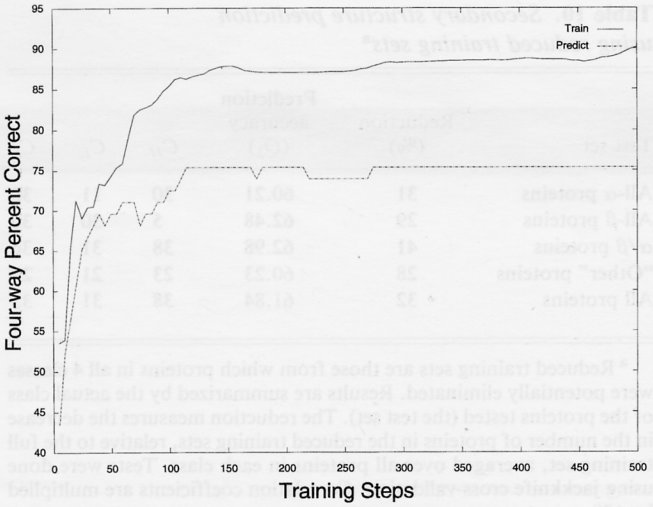

Fig. 4. Class prediction results. The network shown in Figure 2 was tested on all proteins in the data set, using jackknife cross-validation. Training was paused after every 5 steps to measure accuracy on both the training and prediction sets. After 500 steps of training, accuracy on the training set reaches 90%; accuracy on the prediction set levels out at 75%.

Secondary structure prediction with class elimination

Given the results for the class prediction, it appears that the quality of the training set for secondary structure prediction can be improved by the removal of proteins in the class (or classes) eliminated by the class prediction networks. We would expect this to improve accuracy on all-α and all-β proteins, as reported by Kneller et al. (1990) but not for α/β and "other" proteins, where no gain was seen. The following multistage algorithm was used for prediction of the secondary structure and class of a protein of unknown structure:

1. Secondary structure is predicted using the $19 \times 2 \times 2$ network shown in Figure 1, with 1 smoothing step. The training set for the network includes all proteins in a database of known structures.
2. The secondary structure predictions are used to predict the class of the protein using the 4-output network shown in Figure 2 and all 4 single-output networks.
3. If the class is predicted as all-α by any network in step 2, all the all-β proteins are removed from the set; if the class is predicted as all-β in step 2, the all-α proteins are removed from the training set.
4. If proteins have been removed, the secondary structure is predicted again using a $19 \times 2 \times 2$ network trained on the smaller ("reduced") training set.

This algorithm was tested on the 62-protein database, using the jackknife validation method, i.e., each protein chain was removed in turn and the remaining 61 were used as the full training database for the algorithm. The predictions obtained using the full training set were compared with predictions obtained using reduced training sets produced with the above method. Results obtained using the full training set are shown in Table 8, and results obtained using the reduced training sets are shown in Table 9. A graph of training and prediction set accuracy versus training time (for the full and reduced training sets) is shown in Figure 5.

Table 8. *Secondary structure predictions using the full training set*[a]

Test set	Prediction accuracy (Q_3)	C_H	C_E	C_C
All-α proteins	60.87	31	16	32
All-β proteins	61.99	11	33	32
α/β proteins	62.48	38	31	37
"Other" proteins	63.91	37	25	30
All proteins	62.26	37	33	34

[a] Predictions are summarized by the actual class of the proteins tested (the test set). Testing was done using jackknife cross-validation.

Accuracy on both all-α proteins and all-β proteins increased by about 1% when using the reduced training sets. Accuracy on α/β proteins decreased slightly as a result of misclassification of several of these as all-α or all-β. Accuracy on "other" proteins actually increased slightly because several of these contain predominantly helix or sheet and were misclassified into the all-α or all-β classes. The average accuracy on the entire database (weighted by sequence length) increased by 0.38%, from 62.26% to 62.64%. Although these results are less accurate than the best results shown in Table 2, this is a consequence of the variation caused by the selection of a particular set of 48 training proteins and 14 test proteins in those trials; the jackknife procedure used here yields a more unbiased evaluation of prediction accuracy.

We also tested the elimination of α/β and "other" proteins from the training sets, in addition to the removal of all-α and all-β proteins using the algorithm presented above. If the class of a protein was predicted as α/β by any network in step 2, proteins from the "other" class were removed from the training set; if the class was predicted as "other" in step 2, the α/β proteins

Table 9. *Secondary structure prediction using reduced training sets*[a]

Test set	Reduction (%)	Prediction accuracy (Q_3)	C_H	C_E	C_C
All-α proteins	16	62.03	32	17	32
All-β proteins	14	62.99	11	33	31
α/β proteins	5	61.91	38	31	35
"Other" proteins	7	64.69	39	27	30
All proteins	10	62.64	40	33	34

[a] Reduced training sets are those from which all-α and all-β proteins were potentially eliminated. Results are summarized by the actual class of the proteins tested (the test set). The reduction measures the decrease in the number of proteins in the reduced training sets, relative to the full training set, averaged over all proteins in each class, e.g., for proteins in the all-α class, an average of 16% of the proteins in the full training set were eliminated to produce the reduced training sets. Tests were done using jackknife cross-validation. Correlation coefficients are multiplied by 100.

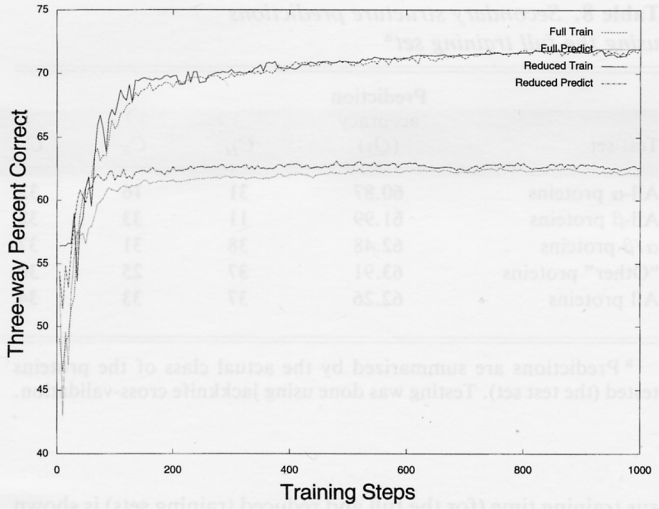

Fig. 5. Reduced versus full training sets. A 19 × 2 × 2 network was tested on all proteins in the data set, using jackknife cross-validation. Training was paused after every 5 steps to measure accuracy on both the training and prediction sets. Results were weighted by sequence length and averaged. Results using the training set reducing algorithm are compared to results using the entire training set. The reduced results consistently remain 0.4–1% higher than the unreduced results over 1,000 training steps.

were removed from the training set. Results obtained using this addition to the reduction algorithm are shown in Table 10.

The modified reduction method led to a decrease in secondary structure prediction accuracy for proteins from most structural classes because elimination of the large α/β proteins could remove many residues from the training set without changing the proportions of secondary structure content. Average accuracy on all 4 classes (weighted by sequence length) decreased by 0.42%.

Table 10. *Secondary structure prediction using reduced training sets*[a]

Test set	Reduction (%)	Prediction accuracy (Q_3)	C_H	C_E	C_C
All-α proteins	31	60.21	30	11	30
All-β proteins	29	62.48	5	30	30
α/β proteins	41	62.98	38	31	36
"Other" proteins	28	60.23	23	21	24
All proteins	32	61.84	38	31	33

[a] Reduced training sets are those from which proteins in all 4 classes were potentially eliminated. Results are summarized by the actual class of the proteins tested (the test set). The reduction measures the decrease in the number of proteins in the reduced training sets, relative to the full training set, averaged over all proteins in each class. Tests were done using jackknife cross-validation. Correlation coefficients are multiplied by 100.

Discussion

A pair of neural network-based algorithms for predicting the secondary structure and structural class of proteins is presented. By using information provided by the secondary structure prediction network, the accuracy of the class prediction network improves by 11.6%, from 62.3% to 73.9%. Using predicted class information, the secondary structure prediction network realizes a small increase in accuracy, from 62.26% to 62.64%. This increase may not be significant.

The structural class prediction results demonstrate that secondary structure prediction, while an interesting theoretical problem in itself, is also useful as a step toward the prediction of aspects of tertiary structure, such as the structural class of a protein. It is important for a tertiary structure prediction algorithm to make use of all other relevant predictions. In the present case, inclusion of single sequence secondary structure predictions improved results by 11.6%. It is possible that the use of a more accurate multisequence profile secondary structure prediction algorithm such as that of Rost and Sander (1993a) would improve this result further.

The multistage secondary structure prediction algorithm also demonstrates the possible benefits of cooperative structure prediction algorithms. Although the accuracy of the class prediction algorithms presented here is too low to reliably narrow the training set down to proteins of a single structural class, the algorithm can, with near-perfect accuracy, eliminate one or more structural classes as a possibility. This limited prediction results in marginal improvements in secondary structure prediction accuracy. Removal of a single class from the training set results in a 1.2% increase in accuracy for all-α proteins (when all-β proteins are removed), compared to a 3% increase in accuracy when all other classes are removed (Rost & Sander, 1993b). This increase in accuracy is not the result of simplifying the secondary structure prediction problem from 3 states to 2 states (i.e., helix or nonhelix for all-helical proteins), as done by Kneller et al. (1990). In fact, prediction of β-strands in these all-α proteins is actually slightly more accurate after reducing the training set (C_E increases from 0.16 to 0.17 for proteins in the all-α class). Further improvements in accuracy could result from the use of larger data sets. The use of the class prediction algorithm eliminates 1/2 to 1/4 of the data in this relatively small data set and so interferes with the ability of the neural network to derive general rules for secondary structure prediction. We are presently extending the approach to larger data sets to investigate this effect.

The smoothing filter applied in the secondary structure prediction algorithm can be a useful tool for reducing noise in the data and slightly improving the accuracy of predictions without the need for a more complex algorithm. This filter is also useful as a visualization tool in viewing the location of helices and sheets in a secondary structure prediction.

Although the work presented has focused on prediction of the secondary structure of single sequences, both the methods of smoothing and training set selection should be applicable to algorithms that operate on a profile of multiple, related sequences such as that used by Rost and Sander (1993a), which is based on a similar network.

Finally, we have confirmed a significant dependence of the results obtained from the neural network algorithms on the choice of training and test sets (Zhang et al., 1992; Rost & Sander, 1993a). Future predictions should use the jackknife strategy of removal of each protein individually from the data-

base to avoid variation in the results caused by a particular choice of training and test sets. If this method is impractical due to the longer time required, multiple cross-validation (several random partitionings of the data set into training and test sets) should be used to eliminate bias. As a side effect, the jackknife method reduces or eliminates the effects of overtraining a network.

Acknowledgment

J.-M.C. is a Howard Hughes Medical Institute Predoctoral Fellow.

References

Argos P. 1987. Analysis of sequence-similar pentapeptides in unrelated protein tertiary structures. *J Mol Biol 197*:331–348.

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

Dubchak IS, Holbrook, Kim S. 1993. Prediction of folding class from amino acid composition. *Proteins Struct Funct Genet 16*:79–91.

Holley H, Karplus M. 1989. Protein secondary structure prediction with a neural network. *Proc Natl Acad Sci USA 86*:152–156.

Holley H, Karplus M. 1991. Neural networks for protein structure prediction. *Methods Enzymol 202*:204–224.

Kabsch W, Sander C. 1983a. How good are predictions of protein secondary structure? *FEBS Lett 155*:179–182.

Kabsch W, Sander C. 1983b. Dictionary of protein secondary structure: Pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers 22*:2577–2637.

Kabsch W, Sander C. 1984. On the use of sequence homologies to predict protein structure: Identical pentapeptides can have completely different conformations. *Proc Natl Acad Sci USA 81*:1075–1078.

Klein P, DeLisi C. 1986. Prediction of protein structural class from the amino acid sequence. *Biopolymers 25*:1659–1672.

Kneller DG, Cohen FE, Langridge R. 1990. Improvements in protein secondary structure prediction by an enhanced neural network. *J Mol Biol 214*:171–182.

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

Matthews BW. 1975. Comparison of the predicted and observed secondary structure of T4 phage lysozyme. *Biochim Biophys Acta 405*:442–451.

Muskal S, Kim S. 1992. Predicting protein secondary structure content: A tandem neural network approach. *J Mol Biol 225*:713–727.

Presnell SR, Cohen BI, Cohen FE. 1992. A segment-based approach to protein secondary structure prediction. *Biochemistry 31*:983–993.

Qian N, Sejnowski TJ. 1988. Predicting the secondary structure of globular proteins using neural network models. *J Mol Biol 262*:865–884.

Rooman M, Wodak SJ. 1988. Identification of predictive sequence motifs limited by protein structure data base size. *Nature 335*:45–49.

Rost B, Sander C. 1993a. Improved prediction of protein secondary structure by use of sequence profiles and neural networks. *Proc Natl Acad Sci USA 90*:7558–7562.

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

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

Rumelhart D, Hinton GE, Williams RJ. 1986. Learning internal representations by error propagation. In: Rumelhart DE, McClelland JL, eds. *Parallel distributed processing, vol 1*. Cambridge, Massachusetts: MIT Press. pp 318–362.

Stolorz P, Lapedes A, Xia Y. 1992. Predicting protein secondary structure using neural net and statistical methods. *J Mol Biol 225*:363–377.

Taylor WR, Thornton JM. 1984. Recognition of super-secondary structure in proteins. *J Mol Biol 173*:487–514.

Zhang C, Chou K. 1992. An optimization approach to predicting protein structural class from amino acid composition. *Protein Sci 1*:401–408.

Zhang X, Mesirov JP, Waltz DL. 1992. Hybrid system for protein secondary structure prediction. *J Mol Biol 225*:1049–1063.