CHAPTER

4

Predicting Protein Structural Features With Artificial Neural Networks

Stephen R. Holbrook, Steven M. Muskal

and Sung-Hou Kim

1. Introduction

The prediction of protein structure from amino acid sequence has become the Holy Grail of computational molecular biology. Since Anfinsen [1973] first noted that the information necessary for protein folding resides completely within the primary structure, molecular biologists have been fascinated with the possibility of obtaining a complete three-dimensional picture of a protein by simply applying the proper algorithm to a known amino acid sequence. The development of rapid methods of DNA sequencing coupled with the straightforward translation of the genetic code into protein sequences has amplified the urgent need for automated methods of interpreting these one-dimensional, linear sequences in terms of three-dimensional structure and function.

Although improvements in computational capabilities, the development of area detectors, and the widespread use of synchrotron radiation have reduced the amount of time necessary to determine a protein structure by X-ray crystallography, a crystal structure determination may still require one or more man-years. Furthermore, unless it is possible to grow large, well-ordered crystals of the protein of interest, X-ray structure determination is not even an option. The development of methods of structure determination by high resolution 2-D NMR has alleviated this situation somewhat, but this technique is also costly, time-consuming, requires large amounts of protein of high solubility and is severely limited by protein size. Clearly, current experimental methods of structure determination will not be able to cope with the present and future need for protein structure determination.

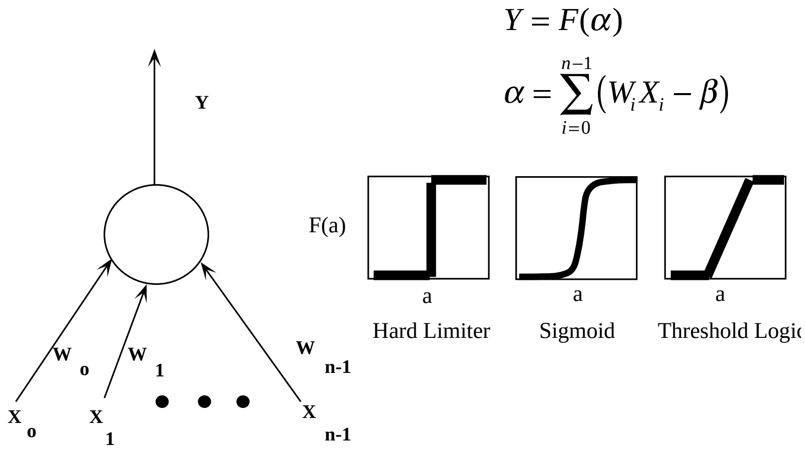
Efforts toward protein structure prediction have come from two general directions and their hybrids. The first, a molecular mechanics approach, assumes that a correctly folded protein occupies a minimum energy conformation, most likely a conformation near the global minimum of free energy. Predictions are based on a forcefield of energy parameters derived from a variety of sources including *ab initio* and semi-empirical calculations and experimental observations of amino acids and other small molecules [Weiner, *et al* 1984]. Potential energy is obtained by summing the terms due to bonded (distance, angle, torsion) and non-bonded (contact, electrostatic, hydrogen bond) components calculated from these forcefield parameters [Weiner & Kollman, 1981]. This potential energy can be minimized as a function of atomic coordinates in order to reach the nearest local minimum. This method is very sensitive to the protein conformation at the beginning of the simulation. One way to address this problem is use molecular dynamics to simulate the way the molecule would move away from that (usually arbitrary) initial state. Newton's equations of motion are used to describe the acceleration of atoms in a protein with respect to time; the movement in this simulation will be toward low energy conformations. The potential energy of the molecule can also be minimized at any point in a dynamics simulation. This method searches a larger proportion of the space of possible confirmations.

Nevertheless, only through an exhaustive conformation search can one be insured to locate the lowest energy structure. Even restricting the representation of a confirmation of a protein as much as possible, to only a single point of interest per amino acid and two angles connecting the residues, the combinatorial aspect of an exhaustive search lead to difficult computational problems [Wetlaufer, 1973]. Under the further simplification of restricting each atom in the protein chain to a discrete location on a lattice [Covell & Jernigan, 1990] and searching the conformation space with very simple energy equations, the exhaustive search method is feasible for only small proteins. Alternatively, conformational space may be sampled randomly and sparsely by monte carlo methods with the hope that a solution close enough to the global energy minimum will be found so that other methods will be able to converge to the correct conformation. Given an approximately correct model from either monte carlo searches or other theoretical or experimental approaches, the technique of molecular dynamics has become the method of choice for refinement, or improvement, of the model. This approach allows the moving molecule to overcome some of the traps of local energy minima in its search for a global minimum.

In general, the energetics approach of molecular mechanics is fraught with problems of inaccurate forcefield parameters, unrealistic treatment of solvent, and landscapes of multiple minima. It appears that this direction will be most valuable in combination with other methods which can provide an approximate starting model.

The second major focus of research toward predicting protein structures from sequence alone is a purely empirical one, based on the databases of known protein structures and sequences. This approach hopes to find common features in these databases which can be generalized to provide structural models of other proteins. For example, the different frequencies at which various amino acid types occur in secondary structural elements; helices, strands, turns and coils, has led to methods [Chou & Fasman, 1974a; Chou & Fasman, 1974b; Garnier, Osguthorpe & Robson, 1978; Lim, 1974a; Lim, 1974b] for predicting the location of these elements in proteins. Even more powerful and now widely used is the prediction of tertiary structure by sequence homology or pattern matching to previously determined protein structures [Blundell, Sibanda & Pearl, 1983; Greer, 1981; Warme, et al, 1974] or structural elements, such as zinc binding fingers, helix-turn-helix DNA binding motifs and the calcium binding EF hand. A portion of a target protein that has a sequence similar to a protein or motif with known structure is assumed to have the same structure. Unfortunately, for many proteins there is not sufficient homology to any protein sequence or sub-sequence of known structure to allow application of this technique. Even proteins thought to have similar structures on functional grounds may show such little sequence similarity that it is very difficult to determine a proper sequence alignment from which to propose a molecular model.

Thus, an empirical approach, which derives general rules for protein structure from the existing databases and then applies them to sequences of unknown structure currently appears to be the most practical starting point for protein structure prediction. Various methods have been used for extracting these rules from structural databases, ranging from visual inspection of the structures [Richardson, 1981], to statistical and multivariate analyses [Chou & Fasman, 1974; Krigbaum & Knutton, 1973]. Recently, artificial neural networks have been applied to this problem with great success [Crick, 1989]. These networks are capable of effecting any mapping between protein sequence and structure, of classifying types of structures, and identifying similar structural features from a database. Neural network models have the advantage of making complex decisions based on the unbiased selection of the most important factors from a large number of competing variables. This is particularly important in the area of protein structure determination, where the principles governing protein folding are complex and not yet fully under-


$$Y = F(\alpha)$$

$$\alpha = \sum_{i=0}^{n-1} \left(W_i X_i - \beta \right)$$

Figure 1: A computational node represented as a circle with weighted inputs and output shown as arrows. The formula for summation of weighted input and bias (b) is given, as well as three common functional forms of nonlinearity which may be used by the node to determine output

stood. The researcher is then able to explore various hypotheses in the most general terms, using the neural network as a tool to prioritize the relevant information.

The remainder of this review will discuss neural networks in general including architecture and strategies appropriate to protein structure analysis, the available databases, specific applications to secondary and tertiary structure prediction, surface exposure prediction, and disulfide bonding prediction. Finally, we will discuss the future approaches, goals and prospects of artificial neural networks in the prediction of protein structure.

2. Artificial Neural Networks

Artificial neural networks appear well suited for the empirical approach to protein structure prediction. Similar to the process of protein folding, which is effectively finding the most stable structure given all the competing interactions within a polymer of amino acids, neural networks explore input information in parallel. . Inside the neural network, many competing hypotheses are compared by networks of simple, non-linear computation units. While many types of computational units exist, the most common sums its inputs and passes the result through some kind of nonlinearity. Figure 1 illustrates a typical computational node and three common types of nonlinearity; hard limiters, sigmoidal, and threshold logic elements. Nearly every neural network model is composed of these types of computational units. The main

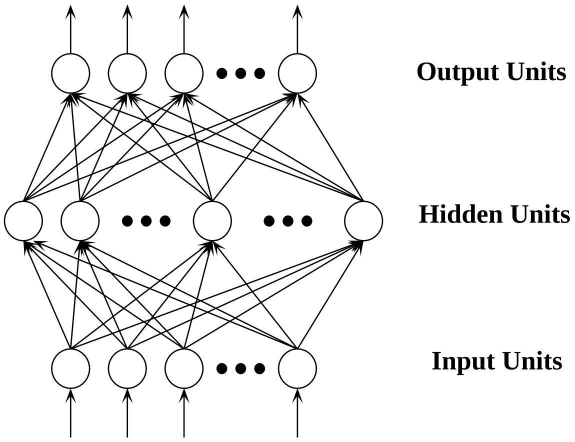


Figure 2. A three layer feedforward neural network. The circles represent the computational nodes which integrate input from the preceding layer and transmit a signal to the next layer. Arrows represent weighted links (connections) between these nodes which modulate incoming signals. The three layer network presented is the most common, but additional layers are possible.

differences exist in topology (node connectivity), methods of training, and application. This article will focus primarily on one type of network, the feedforward network trained with backpropagation for rule extraction purposes. Networks are termed feedforward because information is provided as input and propagated in a forward manner, with each computational unit integrating its inputs and "firing" according to its non-linearity. The following sections will describe in more detail the characteristics of feedforward networks, the preferred method of training with backpropagation, and useful techniques for network optimization.

2.1 Feedforward Networks

A typical feed-forward network is depicted in Figure 2. These networks are often composed of two to three layers of nodes; input and output or input, hidden, and output. Each network has connections between every node in one layer and every other node in the layer above. Two layer networks, or perceptrons, are only capable of processing first order information and consequently obtain results comparable to those of multiple linear regression. Hidden node networks, however, can extract from input information the higher order features that are ignored by linear models.

Feedforward networks are taught to map a set of input patterns to a corresponding set of output patterns. In general, a network containing a large enough number of hidden nodes can always map an input pattern to its corresponding output pattern [Rumelhart & McClelland, 1986]. Once such networks learn this mapping for a set of training patterns, they are tested on examples that are in some way different from those used in training. While most feedforward networks are designed to maximize generalization from training examples to testing examples, some networks are intentionally forced to memorize their training examples. Such networks are then tested with either an incomplete or subtly different pattern. The output of the network will be the memory that best matches the input..

2.2 Training Procedure

The process of training a feedforward network involves presenting the network with an input pattern, propagating the pattern through the architecture, comparing the network output to the desired output, and altering the weights in the direction so as to minimize the difference between the actual output and the desired output. Initially however, the network weights are random and the network is considered to be ignorant. While many algorithms exist for training, clearly the most frequently used technique is the method of backpropagation [Rumelhart, Hinton & Williams, 1986]. Backpropagation involves two passes through the network, a forward pass and a backward pass. The forward pass generates the network's output activities and is generally the least computation intensive. The more time consuming backward pass involves propagating the error initially found in the output nodes back through the network to assign errors to each node that contributed to the initial error. Once all the errors are assigned, the weights are changed so as to minimize these errors. The direction of the weight change is:

$$\Delta W_{ij} = \upsilon \cdot \delta_j \cdot O_i \qquad (1)$$

where W_{ij} is the weight from node *i* to node *j*, ν is a learning rate, δ_j is an error term for node *j*, O_i is either the output of node *i* or an input value if node *i* is an input node. If the node j is an output node, then

$$\delta_j = F_j'\left(net_j\right) \cdot \left(T_j - O_j\right) \qquad (2)$$

with

$$net_j = \sum_i \left(W_{ij} \cdot O_i\right) \qquad (3)$$

where $F_j{}'(net_j)$ is the derivative of the nonlinear activation function which maps a unit's total input to an output value, T_j is the target output of the output node and O_j is the actual output. If node *j* is an internal hidden node, then

$$\delta_j = F_j'\left(net_j\right) \cdot \sum_{k>j} \left(\delta_k \cdot W_{jk}\right) \qquad (4)$$

The weight change as described in Equation 1 can be applied after each example, after a series of examples, or after the entire training set has been presented. Often momentum terms are added and weight changes are smoothed to effect faster convergence times. Regardless of the training recipe however, the main goal of the network is to minimize the total error *E* of each output node *j* over all training examples *p*:

$$E = \sum_{p} \sum_{j} \left(T_j - O_j\right)^2 \tag{5}$$

2.3 Network Optimization

Because the rules in most input-output mappings are complex and often unknown, a series of architecture optimizing simulations are required when testing each hypothesis. Examples of such optimizing experiments include varying input representation, numbers of hidden nodes, numbers of training examples, etc. In each case, some measure of network performance is evaluated and tabulated for each network architecture or training condition. The best performing network is chosen as that which performs the best on both the training and testing sets.

With networks containing hidden nodes, training algorithms face the problem of multiple-minima when minimizing the output error across all training patterns. If the error space is rugged, as is often the case in hidden node networks, the multiple-minima problem can be a serious one. To combat this problem, researchers often permute their training and testing sets and train a number of times on each set, while reporting the best performing network for each simulation. The variance between training and testing sets as well as between training sessions helps to describe the complexity of the weight space as well as the input-output mapping.

Generally smooth trends in performance levels immediately point to optimal network architectures. One nuisance to those who are designing networks to generalize from training examples to testing examples, however, is the concept of memorization or overfitting: the network learns the training examples, rather than the general mapping from inputs to outputs that the training set exemplifies. Memorization reduces the accuracy of network generalization to untrained examples. Sure signs of undesired memorization become apparent when the network performs much better on its training set than on its testing set; and typically, this results when the network contains far more weights than training examples. When undesired memorization results, the researcher is forced to increase the numbers of training examples, reduce node connectivity, or in more drastic situations, reduce the number of input, hidden, and/or output nodes. Increasing the number of training examples is by far the best remedy to the effects of memorization. But more often than not, especially in the area of protein structure prediction, one is constrained with a relatively small database. If it is not possible to increase the database of training examples, the next best choice is to reduce the network connectivity. This, however, poses the problem of deciding on which connections to remove. Here, some have tried removing those connections that are used the least or that vary the most in the training process. This process of network pruning, however, often slows the already lengthy training process and should be done with caution. Finally, reducing the number of network nodes is the least desirable of all approaches since it often results in hiding key information from the network, especially if the number of input nodes is reduced. Similarly, reducing the number of hidden nodes often results in unacceptable input-output mappings; while reducing the number of output nodes, often results in mappings that are no longer useful. Clearly, undesired memorization is one of the greatest drawbacks with neural network computing. Until methods for alleviating the problem are developed, researchers are forced to be clever in their design of representations and network architecture.

Feedforward neural networks are powerful tools. Aside from possessing the ability to learn from example, this type of network has the added advantage of being extremely robust, or fault tolerant. Even more appealing is that the process of training is the same regardless of the problem, thus few if any assumptions concerning the shapes of underlying statistical distributions are required. And most attractive is not only the ease of programming neural network software, but also the ease with which one may apply the software to a large variety of very different problems. These advantages and others have provided motivation for great advances in the arena of protein structure prediction, as the following sections suggest.

2.4 Protein Structure and Sequence Databases

Application of an empirical approach to protein structure prediction is entirely dependent on the experimental databases which are available for analysis, generalization and extrapolation. Since all of the studies discussed below are dependent on these databases, a brief discussion of their contents is appropriate.

The Brookhaven Protein Data Bank [Bernstein *et al*, 1977], or PDB, currently (April, 1990) contains atomic coordinate information for 535 entries. These entries are primarily determined by X-ray crystallography, but some more recent entries are from two-dimensional NMR and molecular modeling studies. Of the 535 entries, 37 are nucleic acids, 10 are polysaccharides and 27 are model structures. Of the remaining entries many of the proteins are essentially duplicated, with either minor amino acid changes due to biological source or specific mutation or with different ligands bound. Taking these factors into account, one can estimate that the Protein Data Bank, currently contains 180 unique protein coordinates sets. Besides the x, y, z coordinates of the non-hydrogen atoms of the proteins and bound co-factors, the following information is included in the Protein Data Bank entries: protein name, a list of relevant literature references, the resolution to which the structure was determined, the amino acid sequence, atomic connectivity, the researcher's judgement of secondary structure and disulfide bonding pattern, and also may contain atomic temperature factors (measure of mobility), coordinates of bound water molecules and other ligands, a discussion of the refinement scheme and its results (estimate of error), and other miscellaneous comments the depositors may wish to make.

In addition to the information directly available from the PDB several computer programs are available both through Brookhaven and from external sources for calculation of additional structural parameters from the entries. These programs calculate such values as the main chain conformational angles phi and psi, the side chain torsion angles, the surface area accessible to a water molecule, distances between all residue pairs in the form of a matrix and may also make automatic assignments of disulfide bonds, secondary structure and even super-secondary structure folding patterns. The most widely used of these programs and the one employed for most of the neural network studies is the DSSP program of Kabsch and Sander [Kabsch & Sander, 1983].

Because of the difficulty of the experimental methods of protein structure determination, the number of known three-dimensional protein structures is much less than the number of protein sequences which have been determined. It is vital, then, to merge this information together with the structural information of the PDB in attempts to predict protein structure. The Protein Identification Resource [George, *et al*, 1986] or PIR, as of December 31, 1989 contained 7822 protein sequences consisting of 2,034,937 residues. The amino acid sequences of these proteins were determined either by chemical sequencing methods or inferred from the nucleic acid sequences which code for them. The PIR database contains, in addition to amino acid sequence, information concerning the protein name, source, literature references, functional classification and some biochemical information.

An even larger database of sequences is found in the GENBANK collection of nucleic acid sequences. Many of these sequences code for proteins whose sequences may be obtained by a simple translation program. The nucleic acid sequences which code for proteins may eventually become the source for additional entries in the PIR, but because of the rapid growth of both the GENBANK and PIR databases there currently is a large backlog of sequences to be added to these data banks.

A variety of computer programs also are available for analysis of the protein sequence database, the PIR. These programs include those which calculate amino acid composition, search for sequence similarity or homology, conserved functional sequences, plot hydrophobicity and predict secondary structure.

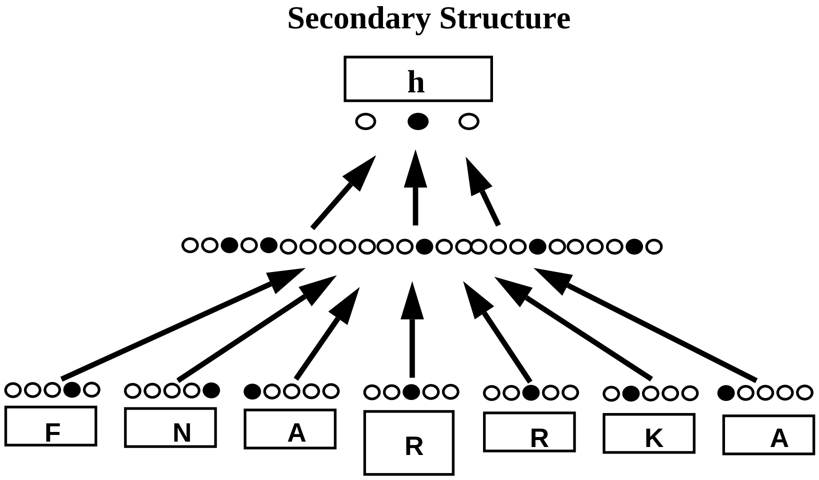


Figure 3: A feedforward neural network of the type used by Qian and Sejnowski [1988] for the prediction of secondary structure from a window of input amino acid sequence. Active nodes are shaded and the connections between each node and all other nodes above it are illustrated schematically by arrows. Only 5 input nodes are shown for each amino acid although 21 were used.

3. Secondary Structure Prediction with Neural Networks

At present, the largest application of feedforward neural networks in the world of protein structure prediction has been the prediction of protein secondary structure. As secondary structures (α-helices, β-strands, β-turns, etc) are by definition the regions of protein structure that have ordered, locally symmetric backbone structures, many have sought to predict secondary structure from the sequence of contributing amino acids [Chou & Fasman, 1974a; Chou & Fasman, 1974b; Garnier, Osguthorpe & Robson, 1978; Lim, 1974a; Lim, 1974b[. Recently though, Qian and Sejnowski (1988], Holley and Karplus [1989], Bohr *et al*. [1988], and McGregor *et al*. [1989] have applied neural network models to extract secondary structural information from local amino acid sequences and have achieved improved secondary structure prediction levels over that derived by statistical analysis [Chou & Fasman, 1974a; Chou & Fasman, 1974b].

3.1 α-Helix, β-Strand, and Coil Predictions

The general hypothesis taken when attempting to predict secondary structure is that an amino acid intrinsically has certain conformational preferences and these preferences may to some extent be modulated by the locally surrounding amino acids. Using this information, network architectures of the

Window Size	Q_3(%)	C_α	C_β	C_{coil}
1	53.90	0.11	0.14	0.17
3	57.70	0.22	0.20	0.30
5	60.50	0.28	0.26	0.37
7	61.90	0.32	0.28	0.39
9	62.30	0.33	0.28	0.38
11	62.10	0.36	0.29	0.38
13	**62.70**	**0.35**	**0.29**	**0.38**
15	62.20	0.35	0.31	0.38
17	61.50	0.33	0.27	0.37
21	61.60	0.33	0.27	0.32

Table 1: Dependence of testing accuracy on window size (adapted from Qian & Sejnowski, 1988). Q_3 is average percent correct over three predicted quantities (α, β, coil). C is correlation coefficient for each prediction type, as defined by Mathews [1975].

type in shown in Figure 3 have been designed to predict an amino acid's secondary structure given the sequence context with which it is placed.

Qian and Sejnowski [1988] and others [Holley & Karplus 1989; Bohr *et al*. 1988] have shown that a locally surrounding window of amino acids does improve prediction levels as shown in Table 1. This table indicates that when the size of the window was small, the performance on the testing set was reduced, suggesting that information outside the window is important for predicting secondary structure. When the size of the window was increased beyond 6 residues on each side of a central residue, however, the performance deteriorated. Therefore, when using only local sequence information, residues beyond 6 residues in each direction contribute more noise than information in deciding a central amino acid's secondary structure.

Further attempts at improving prediction levels by adding a variable num-

Hidden Units	Q_3(%)
0	62.50
5	61.60
10	61.50
15	62.60
20	62.30
30	62.50
40	62.70
60	61.40

Table 2: Testing of secondary structure prediction versus number of hidden nodes. (adapted from Qian & Sejnowski, 1988)

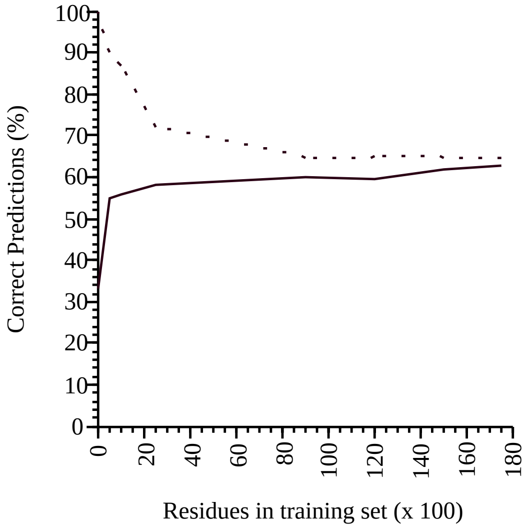


Figure 4: Relationship between prediction accuracy on the Training and Testing sets and number of residues in the Training set. Adopted from Qian and Sejnowski [1988]i

ber of hidden nodes as seen in Table 2 were only slightly successful. In fact, the best performing network containing 40 hidden nodes offers only a small improvement over the network containing 0 hidden nodes. This result suggests that the mapping between flanking amino acid sequence and an amino acid's secondary structure is of first order, requiring little if any higher order information (information due to interactions between 2 or more residues in the input sequence).

Further studies showed the maximum performance of the network as a function of the training set size as seen in Figure 4. The maximum performance on the training set decreases with the number of amino acids in the training set because more information is being encoded in a fixed set of weights. The testing set success rate, however, increases with size because the larger training set increases the network's generalization ability. Figure 4 nicely depicts the concept of memorization. When the training set is small, the network can memorize the details and suffers on the testing set. When the training set is large, memorization is not possible and generalization is forced. Furthermore, Figure 4 suggests that any additional increase in the size of the training set is unlikely to increase the network's testing performance, implying that more information for predicting secondary structure is required than that contained in a window of 13 consecutive amino acids. This missing information is undoubtedly in the tertiary contacts between residues in the proteins. The three-dimensional fold of the protein chain en-

Method	Q_3(%)	C_α	C_β	C_{coil}
Chou-Fasman	50.00	0.25	0.19	0.24
Garnier	53.00	0.31	0.24	0.24
Lim	50.00	0.35	0.21	0.20
Qian & Sejnowski - 1	62.70	0.35	0.29	0.38
Qian & Sejnowski - 2	64.30	0.41	0.31	0.41
Holley & Karplus	63.20	0.41	0.32	0.36

Table 3: Accuracy comparison of methods of secondary structure prediction. Qian & Sejnowski - 1 is their perceptron network, Qian & Sejnowski - 2 includes a smoothing network using predictions from the first network as input. See text.

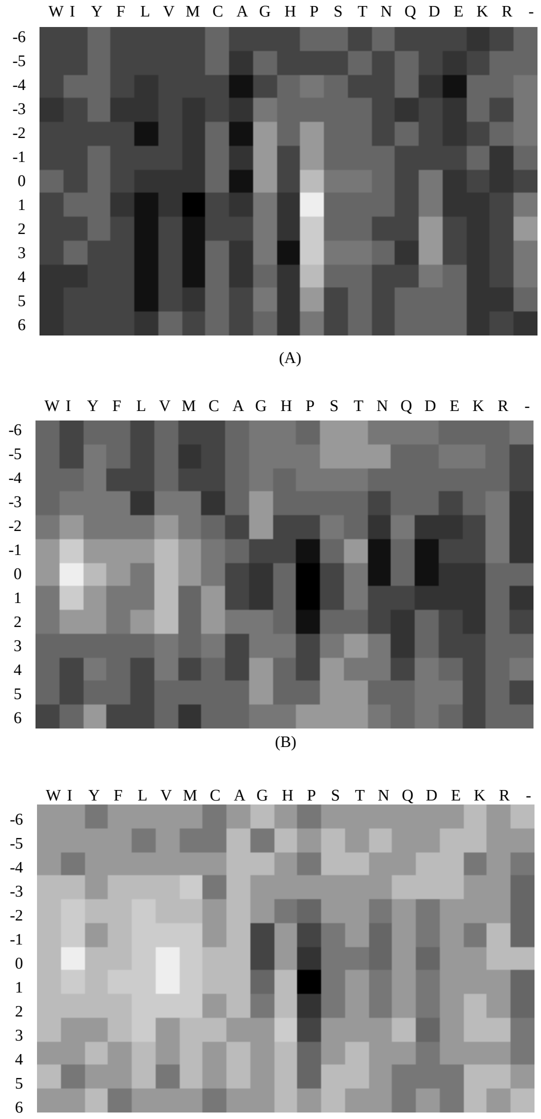
velopes most of the amino acids in a unique environment, thus modifying their inherent tendencies toward a particular secondary structure. A prediction limit is therefore approached when only local sequence information is available.

The performance of Qian and Sejnowski's network compared to those prediction methods of Garnier *et. al.* [1978], Chou & Fasman [1974b], Lim [1974], and Holley & Karplus [1989] is shown in Table 3. Clearly, the neural networks out-perform those methods of the past. Approximately 1% of the 11% improvement in Table 3 between Garnier's method and the neural network method is attributed to the difference between the network's training set and the set of proteins used to compile Garnier's statistics.

One benefit of using networks containing no hidden nodes is the ease with which the network weights can be interpreted. While Sanger [Sanger, D., Personal Communication] has developed a method of weight analysis for hidden node networks called contribution analysis, the technique is still in its infancy. Until more researchers turn to this or other methods of hidden node network weight analysis, graphical representations of the weights from input to output nodes will have to suffice.

Figure 5 details the relative contribution to the decision of a secondary structure made Qian and Sejnowski's network for each amino acid at each window position. Here, correlations between each amino acid's sequence specific secondary structure preference and its physical properties can be readily extracted.

In a parallel study to that of Qian and Sejnowski, Holley and Karplus [1989] have designed a similar network for prediction of secondary structure. Their optimal network contains an input layer of 8 amino acids on either side of the residue of interest (window size equals 17), a hidden layer of two nodes and an output layer of two nodes. The two node output layer describes three states: helix, strand and coil by taking on values of 1/0, 0/1 and 0/0 respectively. Since the actual values of these nodes lie between 0 and 1, a cutoff value or threshold was determined which optimized the network predic-



(A)

(B)

(C)

Figure 5: The relative values of the connection weights obtained by Qian and Sejnowski [1989] in their perceptron network for prediction of helix (a), strand (b) and coil (c) from amino acid sequence. For each window position and amino acid type the weight of its link to the next layer is represented as a shade of gray. Darker shades indicate higher weights. The amino acid residues in this and following similar figures are in order of decreasing hydrophobicity according to Eisenberg [1984]

tion. The maximum overall prediction accuracy on the training set was 63.2% (Table 3) over three states with C_α 0.41, C_β 0.32 and C_{coil} 0.36 which are very similar to the results discussed previously. They also noted an increase in prediction accuracy for residues near the amino-terminus and for highly buried versus partially exposed β-strands. Finally, residues with higher output activities were found to be more accurately predicted, i.e. the strongest 31% of predictions were 79% correct. The Holley and Karplus perceptron network has recently been implemented on an IBM-compatible microcomputer and shown to reproduce their results [Pascarella & Bossa, 1989].

Attempting to extend these studies, Bohr *et al.* [1988] designed three separate networks to predict simply if a residue was in a helix or not, strand or not, and coil or not given a window of 25 residues on each side of a central amino acid. Clearly, by the size of this network, memorization was inevitable. But they, as will be mentioned in their approach to tertiary structure prediction, seem to desire memorization. In fact, their approach seems to have led to a new measure of homology.

Again using a window of 25 residues on each side of a central amino acid, but extending the output to α-helix, β-strand, and coil, Bohr *et al*. trained a network similar to Qian and Sejnowski's on one member of a homologous pair of proteins. The percent performance on the other protein, then, indicated the degree of homology. In this way, Bohr *et al*. used to their advantage the concept of network memorization to determine the degree of similarity between protein sequences, without requiring any sequence alignment.

In a practical application of neural networks for the prediction of protein secondary structure, a prediction of helix and strand location was made for the human immunodeficiency virus (HIV) proteins p17, gp120 and gp41 from their amino acid sequences [Andreassen, *et al*, 1990]. The input layer used an amino acid sequence window of 51 residues (1020 binary units) and a hidden layer of 20 units. Separate networks were trained for α-helices and β-strands and used in their prediction.

3.2 β-turn Predictions

In order for proteins to be compact, folded structures that pack their secondary structures into remarkably small volumes [Richardson, 1981; Rose, 1978], they must have a number of chain reversals. β-Turns are a specific class of chain reversals localized over a four-residue sequence[Richardson, 1981; Venkatachalam, 1968] and are defined by having a distance between Cα(i) and Cα(i+3) of < 7A. Seven classes (I,I',II,II',VIa,VIb,VIII) and a miscellaneous category (IV) have been defined [Richardson, 1981; Venkatachalam, 1968; Lewis, Momany & Sheraga, 1973] and differ by hydrogen bond interactions between involved residues. The most common classes of turns being I and II (41 and 26% of all turns), for example, have a specific

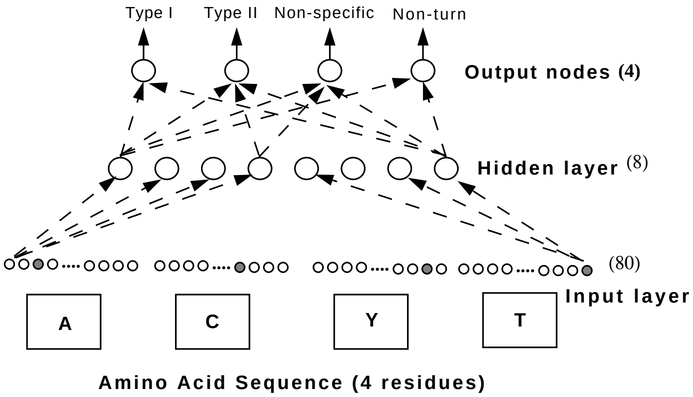


Figure 6. The network architecture used by McGregor, et al. for identification of β-turns. The input layer is a sequence of 4 amino acids comprising aβ-turn or non-turn presented to the network as 20 nodes per amino acid. The output layer has one node per turn (or non-turn) type. Shaded circles indicate activated nodes and dashed arrows schematically represent the weighted links between all node.

hydrogen bond interaction between the C=O of residue i and the N-H of residue i+3.

Similar to the prediction of α-helices and β-strands, network predictions for β-turns begin with the hypothesis that the information necessary to force the sequence of amino acids into a β-turn exists locally in a small window of residues. The network architecture designed to further this notion is depicted in Figure 6. Once again, the input to the network encodes a string of amino acids. The output classifies the sequence as one of four types, Type I, Type II, Non-Specific, or Non-turn.

Because the window size is fixed at four by the definition of β-turns, the only network optimizing simulations required were those that determine optimal numbers of hidden nodes. McGregor *et al.* [1989] have reported, as shown in Table 4 a network performance with 0 (perceptron) and 8 hidden nodes. Statistics were calculated for six different testing sets and the mean value is indicated. Table 4 also compares the performance of these networks to the method of Chou and Fasman [1974b]. The low values for the overall prediction accuracy reflect the stringent requirement that all four residues in the β-turn must be correctly predicted. On an individual residue basis, 71% of the predictions are correct compared to a chance level of 58%.

A commonly occurring issue addressed in this paper is how to adjust the relative ratio of the four different turn types (different outputs) in the training set. Since the numbers of types of turns and non-turns differ considerably, it was important to decide how frequently to sample each input type. Sampling

Prediction Method	% correct	$C_{\beta\text{-turn}}$
Perceptron	24.1	0.177
Hidden Layer Network	26.0	0.202
Chou-Fasman	20.6	0.167

Table 4: Statistics for β-turn prediction

of each type with equal frequency led to a large overdetermination of turns, however if the sequences were sampled according to the frequency at which they actually occur then all the predictions were for non-turns. The authors finally used a trial and error approach, obtaining the best results by sampling type I, II, non-specific turns and non-turns in the ratio 6:3:8:34, approximately the correct ratio except that the non-turns were reduced by a factor of six. This biased distribution of examples may partially account for the low prediction performance obtained with this network.

3.3 Secondary Structure Composition Predictions

Given the above mentioned work, it appears that the information encoded in small windows of local sequence is sufficient to correctly predict approximately two-thirds of a protein's secondary structure [Qian & Sejnowski, 1988; Holley & Karplus, 1989; McGregor, et al, 1989]. Because of this less than satisfactory rate of prediction, many have sought to improve the accuracy of secondary structure predictions by adjusting predictions based on a consensus of many predictive methods [Nishikawa & Ooi, 1986], the secondary structure of seemingly similar proteins [Nishikawa & Ooi, 1986; Levin & Garnier, 1988; Zvelebil, *et al*, 1987], and an *a priori* knowledge of secondary structure composition [Garnier, *et al,* 1978]. In attempts to predict the latter, others have noted that there exists a correlation between secondary structure composition and amino acid composition [Crick, 1989; Nishikawa & Ooi, 1982; Nishikawa, *et al*, 1983].

Neural networks have recently been applied by Muskal and Kim [1992] to the problem of mapping amino acid composition to secondary structure composition. They trained a network to map a string of real numbers representing amino acid composition, molecular weight and presence or absence of a heme cofactor onto two real valued output nodes corresponding to percent α-helix and percent β-strand. A second, or tandem, network was used to detect memorization and maximize generalization.

Networks with and without hidden nodes were able to accurately map amino acid composition to secondary structure composition. The correlations between predicted and real secondary structure compositions for the networks containing no hidden nodes are quite similar to those obtained by techniques of multiple linear regression [Krigbaum & Knutton, 1973; Horne, 1988] and by standard statistical clustering methods [Nishikawa & Ooi, 1982; Nishikawa, et al, 1983], while those obtained with hidden node networks are considerably greater.

The improved performance with networks containing hidden nodes is likely a result of the information contained in combinations of the quantities of each amino acid type, i.e. x amount of Ala with y amount of His. Perhaps secondary structure content is dependent both on composition individual amino acids and on combinations of these compositions. Therefore, in the interest of *de novo* and secondary structure design, serious consideration of potential protagonist and/or antagonist amino acid composition combinations may lead to improved success rates.

The hidden node network's high accuracy, however, (within ±5.0% and ±5.6% for helix and strand composition respectively) is the best predictive performance for secondary structure composition to date and can be attributed to the non-linear mapping of multi-layer neural networks. It should be noted that the error in these predictions is comparable to the errors associated with the experimental technique of circular dichroism (Johnson, 1990).

Utilizing the network weights made available from Qian and Sejnowski [1988] and counting secondary structure predictions, total average errors for helix, strand, and coil composition were approximately ±9.1%, ±12.6%, and ±12.9% respectively. By correcting for predicted secondary composition, Qian and Sejnowski's predictions can be altered to improve the prediction rate from 64% to 67%. Clearly, though secondary structure composition predictions are useful and can offer some improvement to secondary structure prediction, secondary structure predictions do appear to have reached a plateau. This leveling of secondary structure predictions has inspired more effort in the direction of predicting tertiary interactions, as the next sections will suggest.

4. Prediction of Amino Acid Residues on the Protein Surface

The residues on a protein surface play a key role in interaction with other molecules, determine many physical properties, and constrain the structure of the folded protein. Surface exposure of an amino acid residue can be quantified as the area accessible to a water molecule in the folded protein [Lee & Richards, 1971]. The calculation of solvent accessibility, however, has generally required explicit knowledge of the experimentally determined three-dimensional structure of the protein of interest.

Recently, Holbrook, *et al* [1990] have applied neural network methods to extract information about surface accessibility of protein residues from a database of high-resolution protein structures. Neural networks of the type seen in Figure 7 were trained to predict the accessibility of a central residue in context of its flanking sequence.

In order to predict surface exposure of protein residues, it is first neces-

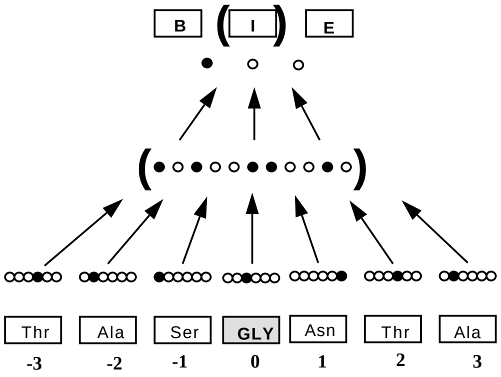


Figure 7. Neural network architecture used for the prediction of solvent accessibility of amino acid residues in proteins. Each amino acid in the window was represented by activating one of 21 binary input nodes. The output consisted of either one, two, or three nodes, corresponding to either a continuous, binary (buried/exposed) or ternary (buried/intermediate/exposed) definition of accessibility

sary to define categories for the buried and exposed residues. Recent definitions [Rose, *et al*, 1985] use the fractional exposure of residues in folded proteins compared with a standard, fully exposed state such as found in extended tripeptides. In the network analysis, two definitions of surface accessible residues were used: 1) a binary model in which buried residues are defined as those with less than 20% of the standard state exposure and accessible residues as those greater than 20% fully exposed and 2) a ternary model in which a residue is either fully buried (0-5% exposure), intermediate (5-40%) exposure, or fully accessible (greater than 40% exposure). A continuous model, which required prediction of the actual fractional exposure was also explored.

The neural networks used in this study contained either zero (perceptron) or one hidden layers and weights set by backpropagation (see Figure 7). The protein sequences were presented to the neural networks as *windows,* or subsequences, of 1-13 residues centered around and usually including the amino acid of interest, which slide along the entire sequence. For experiments involving only the flanking residues, the central residue was omitted from the window.

4.1 Binary Model

Window size was varied between 1 (no neighbors) and 13 (6 amino acids on either side of the central) residues for both training and testing networks containing two outputs. Table 5 shows the results of these experiments. The correct overall prediction for the training set is seen to reach a maximum of about 74% at window size 11 (-5:5) with a correlation coefficient of 0.48. The highest percentage of correct prediction, 72%, and correlation coefficient, 0.44, for the testing set was obtained with a window size of 9 (-4:4) residues. This is only a 2% increase over the 70% obtained with networks trained on patterns of only single amino acids (window size 1). To investigate the significance of this difference and the influence of flanking residues on exposure or burial of the central residue a network using examples consisting of *only* the flanking residues and excluding the central residue was trained and tested on the same databases. This network was able to predict exposure of the central residue in 55.3% of the cases with a correlation coefficient of 0.10 indicating that the sequence of the flanking residues has a small, but significant effect on exposure of the central residue.

Analysis of the predictive capacity of the trained network as a function of location of the residue being predicted in the protein sequence indicated that the residues at the extreme N-terminus can be predicted with much greater accuracy than the protein as a whole. The 10 amino terminal residues of the proteins in the testing set can be correctly predicted in 84% of the cases (correlation coefficient 0.50). A similar, but smaller effect is seen for the residues at the carboxy-termini where 75% of the predictions are correct (correlation coefficient 0.47). The high predictability of the N-terminal residues may reflect the fact that this is the first region of the protein synthesized and as such exists transiently in a different environment from the remainder of the protein. It should also be noted that both the N-terminal and C-terminal portions of the chain are more hydrophilic than the bulk of the protein.

An advantage of neural network analysis is that a prediction of surface exposure is based on quantitative activity values at each of the output nodes. Therefore a confidence level may be assigned to each prediction based on the strength of the output activities. While the accuracy of prediction increases with the minimum activity accepted, a corresponding decrease is seen in the percent of the total residues whose accessibility is predicted. For example, using the binary model of accessibility, while 100% of tested residues are predicted with an accuracy of 72%, over half of the residues with the strongest activities are predicted with greater than 80% accuracy.

4.2 Ternary Model

The use of a three state exposure model offers several advantages over the two state model. First, the definition of buried and exposed residues is clarified since intermediate cases are classified as a third category. Second, it is possible to reproduce the observed distribution more closely by allowing more classes. Finally, if it is not necessary to distinguish between fully and partially exposed residues, it is possible to predict exposure with very high accuracy. In experiments involving three-state prediction (buried, partially exposed, and fully exposed), window size was from 1 to 9 residues, at which

Window Size	%Correct Train Binary	%Correct Test Binary	%Correct Train Ternary	%Correct Test Ternary
1	69.1	70.0	49.1	50.2
3	70.1	69.5	52.4	51.1
5	71.0	70.8	54.1	50.1
7	71.9	71.8	**55.9**	**52.0**
9	**72.5**	**72.0**	57.5	49.8
11	73.9	71.8	-	-
13	73.4	70.7	-	-

Table 5: Solvent exposure predictions

point prediction of the testing set began to decrease. Table 5 gives the results of these experiments for both the training and testing datasets. For both datasets, the fully buried and exposed residues are predicted with greater accuracy than the partially exposed residues As in the experiments with a binary representation, the exposed residues in the testing set are consistently predicted approximately 10% more accurately than the buried. The overall peak in prediction with the ternary model occurs for the testing set at window size 7 (-3:3) after which a decline occurs. Experiments with networks containing a hidden layer of computational nodes between the input and output layers resulted in an improvement in prediction for window size 7 and three output states. The maximal improvement was observed when using 10 hidden nodes, which predicted the testing set with 54.2% overall accuracy, compared to the best prediction of 52.0% with a perceptron network.

Using this three state network with hidden nodes, a residue which is predicted to be fully exposed was actually found to be fully or partially exposed over 89% of the time, while a residue predicted to be buried was found fully or partially buried in 95% of the cases. The difference in prediction percentage for buried and exposed is in large part due to overprediction of the fully exposed state and underprediction of the fully buried state by the network. If only fully exposed or fully buried residues are considered (cases observed or predicted to be partially exposed are ignored) the states are predicted correctly for 87% of the residues. The hydrophobic residues were predicted with very high accuracy (86-100%) as are the hydrophilic residues (75-100%). The ambiphilic residues glycine and threonine were, as expected, predicted with less accuracy (68% and 60% respectively), but the ambiphilic residues methionine, alanine and histidine are predicted with 90-100% accuracy. Even the hydrophobic residue valine is correctly predicted to be exposed in one case and the hydrophilic residue proline is predicted correctly to be buried in one case.

4.3 Continuous Model

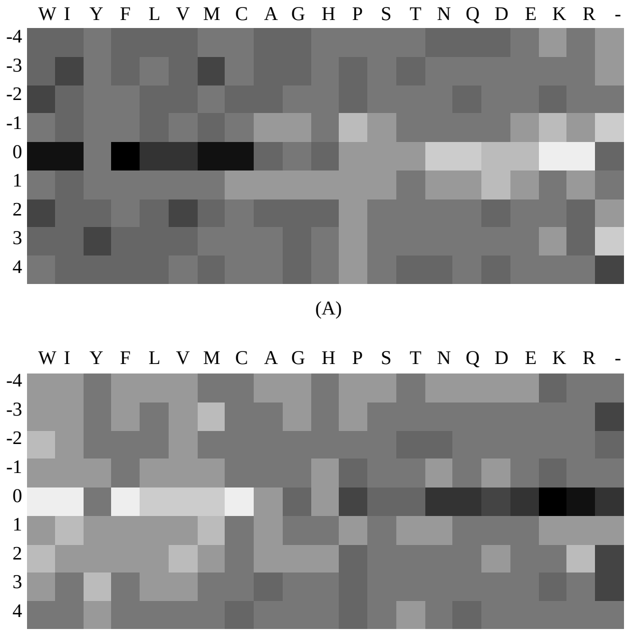
In order to assess the potential for prediction of the percent of fractional exposure without regard to arbitrary definitions of burial and exposure, a direct mapping can be effected from amino acid sequence represented in a binary form as described above (21 nodes per residue) to fractional exposure (S. Holbrook, unpublished results). This mapping utilized real numbers (the actual or predicted fraction exposures of the central residue) as the output nodes which are fit in the training process. Using a window size of 9 amino acid residues, the training set converged at a correlation coefficient of 0.561 with an average deviation between observed and calculated exposure of 17%. This trained network was able to reproduce the exposures of the residues in the testing set with a correlation coefficient of 0.508 and average deviation of 18%.

4.4 Analysis of Network Weights

Examination of the network weights allowed the physical interpretation of the major factors influencing residue exposure. From the plot of network weights in the binary model shown in Figure 8, it is apparent that the primary factor governing exposure of the strongly hydrophobic and hydrophilic residues is the identity of the central amino acid itself, however for neutral or ambiphilic residues such as proline and glycine the flanking sequence is more influential. Nevertheless, the weights show that hydrophobic residues 2 or 3 amino acids before or after the central amino acid favor its burial. This is likely due to the preponderance of buried residues in β-strand and to a lesser degree α-helical structures and the periodicity of these structures. Since exposed residues are favored over buried in turn and coil regions, exposure of the central residue is favorably influenced by neighboring residues such as proline and glycine which preferentially are found in these regions. As turns and coils are not periodic structures, less positional specificity is observed for the exposed residues than for buried residues which prefer regular secondary structure.

The weights to the output nodes of the three state model show a greater contribution of neighboring residues to the exposure of the central residue, especially for the intermediate (partially exposed) node, which is not strongly determined by the central residue alone (not shown). The weights (not shown) suggest that larger residues (i.e. W, H, Y and R) tend towards intermediate exposure (correlation coefficient 0.35) regardless of their hydrophobicity. Generally, high weights for neighboring hydrophobic residues tend to favor burial of the central residue and high weights for neighboring hydrophilic residues favor exposure of the central residue.

In summary, neural network models for surface exposure of protein residues make highly accurate predictions of accessibility based solely on the



(A)

(B)

Figure 8. Network weights for binary model of surface exposure.. (a) is the weight matrix for the buried residue predictions, and (b) is the matrix for the exposed residue predictions.

identity of the amino acid of interest and its flanking sequence. This capability is a valuable tool to molecular biologists and protein engineers as well as to those concerned with the prediction of protein structure from sequence data alone.

5. Prediction of Cysteine's Disulfide Bonding State

The bonding states of cysteine play important functional and structural roles in globular proteins. Functionally, cysteines fix the heme groups in cytochromes, bind metals in ferredoxins and metallothioneins, and act as nucleophiles in thiol proteases. Structurally, cysteines form disulfide bonds that provide stability to proteins such as snake venoms, peptide hormones, immunoglobulins, and lysozymes.

Because free thiols are unstable relative to S-S bridges in the presence of oxygen, cysteines are typically oxidized into disulfide bonds in proteins leaving the cell; and conversely, because S-S bridges are unstable relative to free

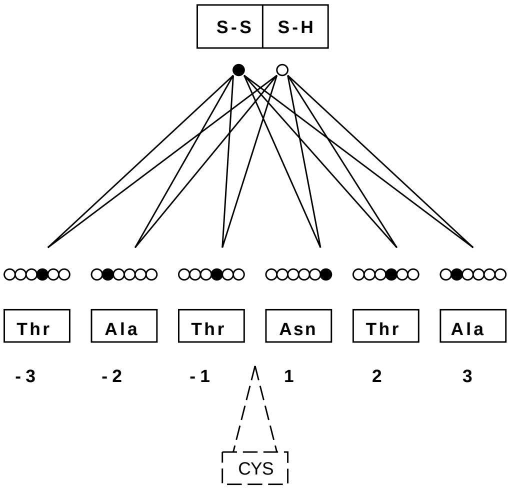


Figure 9. The cysteine network architecture. For clarity, only 6 window positions (3 amino acids to the N-terminal and 3 amino acids to the C-terminal side of the omitted centered cysteine) and 6 nodes per window position are illustrated. The net is a perceptron with two output nodes, one for disulphide bonded cysteines (S-S) and one for hydrogen bonded (S-H).

thiols in reducing environments, cysteines are typically reduced in proteins that remain inside the cell. Predictions of the disulfide bonding state of cysteines based only on this criterion, however, result in failures for extracellular proteins containing free thiols such as actinidin, immunoglobulin, papain, and some virus coat proteins and for cystine containing intracellular proteins such as trypsin inhibitor, thioredoxin, and superoxide dismutase. Furthermore, to base positive disulfide bond predictions on high cysteine content and even parity result in failures for ferredoxins, metallothioneins, and some cytochromes. Clearly, predictions based on these simple rules fail to capture the unique micro-environments a protein structure imposes on its cysteines to define their disulfide bonding states.

Recently, Muskal *et al*. [1990] used a network of the architecture seen in Figure 9 to predict a cysteine's disulfide bonding state, with the presumption that it is the local sequence that influences a cysteine's preference for forming a disulfide bond. The networks were of the feedforward type containing no hidden nodes (perceptrons). Because every sequence presented to the networks contained a centered cysteine, the input layer encoded a window of amino acid sequence surrounding but not including, the central cysteine, as shown in Figure 9

Network performance depended on the size of the window around a centered cysteine. For testing, 30 examples were randomly selected (15 exam-

Window	%Train	$C_{\text{ss-bond}}$	%Test	$C_{\text{ss-bond}}$
-1:1	65.7	.30	60.0	.22
-2:2	72.8	.45	66.7	.34
-3:3	79.1	.57	73.3	.51
-4:4	83.9	.67	73.3	.48
-5:5	85.7	.71	**80.0**	**.61**
-6:6	88.2	.76	80.0	.60
-7:7	91.4	.82	80.0	.61

Table 6: Dependence of training and testing success of the cysteine net on window size. Window of –x:x has x amino acids on either side of the cysteine. C's are Mathews [1975] correlation coefficients.

Run	%Correct Train		%Correct Test	
	S-S	S-H	S-S	S-H
1	89.7	83.3	80.0	80.0
2	89.4	82.3	80.0	80.0
3	89.7	83.3	90.0	70.0
4	90.2	83.0	70.0	90.0
5	90.5	83.0	70.0	100.0
6	90.5	84.3	90.0	70.0
7	90.0	82.7	90.0	70.0
Average	90.0	83.1	81.4	80.0

Table 7: Cross validation runs for cysteine network with window –5:5.

ples of sequences surrounding disulfide bonded cysteines; 15 examples of sequences surrounding non-disulfide bonded cysteines) from the pool of 689 examples, leaving the remaining 659 examples for a training set. The influence of flanking sequence on a centered cysteine was determined by increasing window of sequence surrounding the cysteine and tabulating the network's predictive performance. As seen in Table 6, the network's performance on both the training and testing sets increases with increasing window size. It should be noted that after window -7:7 (14 flanking amino acids, 21 nodes per amino acid, 2 output nodes, and 2 output node biases corresponds to 14 * 21 * 2 + 2 = 590 weights), the number of weights begins to exceed the number of training examples. As a result memorization becomes apparent after a window of -6:6, suggesting that the windows -5:5 or -6:6 are optimal for predictive purposes. Furthermore, Table 6 shows that trained networks made accurate predictions on examples never seen before thus supporting the hypothesis that a cysteine's propensity and/or aversion for disulfide bond formation depends to a great extent on its neighbors in sequence.

Network performance for each set was evaluated by testing on a random

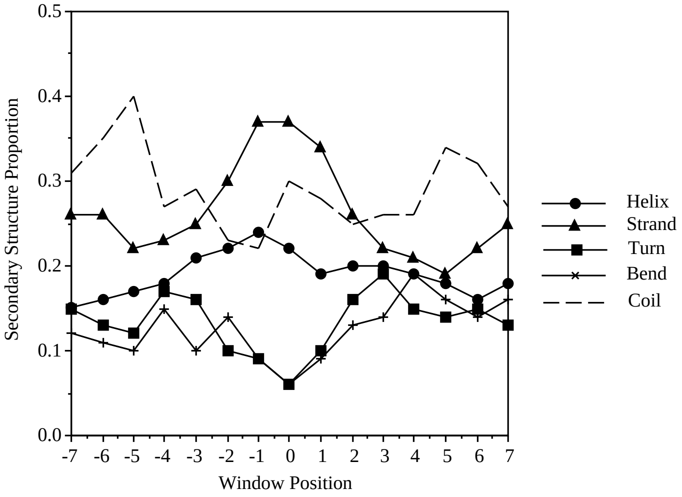
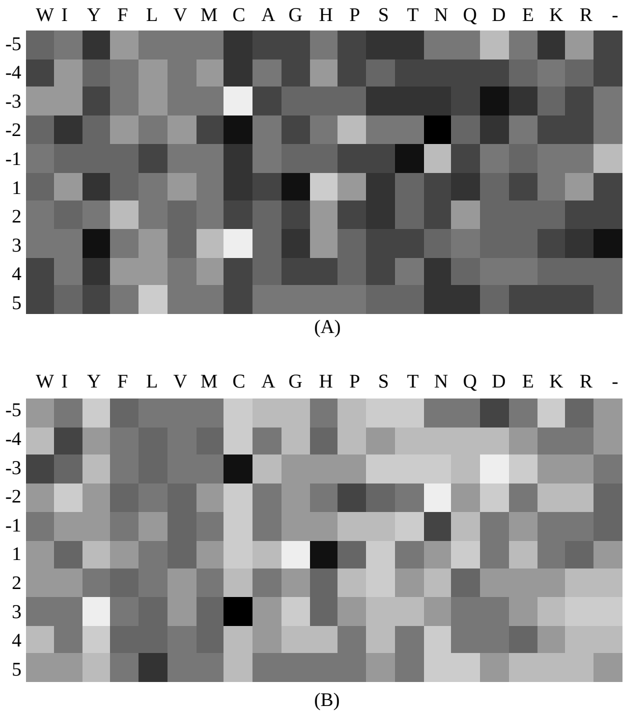


Figure 10. Secondary structure surrounding disulfide bonded cysteines. Secondary structure proportion is calculated by summing number of individual secondary structure types and dividing by the total number of secondary structure occurring in that window position. Secondary structure assignments were made by the method of Kabsch and Sander [1983].

subset of 20 examples (10 examples of sequences surrounding disulfide bonded cysteines; 10 examples of sequences surrounding non-disulfide bonded cysteines) taken from the pool of 689 examples after training on the remaining 669 examples. Each experiment was conducted independently on networks with a window -5:5 (5 amino acids to the left and 5 to the right of a central cysteine).

After window size experiments were completed, 7 independent training and testing experiments were conducted so as to determine an average performance that was not dependent on any particular training and testing set. Table 7 indicates that a network can be trained to predict disulfide bonded scenarios 81.4% correctly and non-disulfide bonded scenarios 80.0% correctly. Trained networks made accurate predictions on sequences from both extracellular and intracellular proteins. In fact, for the extracellular proteins actinidin, immunoglobulin, and papain, the odd cysteines not involved in disulfide bonds were correctly predicted as such. Likewise, for the intracellular cystine-containing proteins such as trypsin inhibitor and superoxide dismutase, every cysteine's state was correctly predicted.

Figure 10 shows the secondary structure proportion as a function of window position for disulfide bonded cysteines. Here the sequences surrounding and including half-cysteines seem to prefer the extended conformation of β–sheets over that of turns and bends. The secondary structural preferences of



(A)

(B)

Figure 11. Weights for the connections to the S-S (a) and S-H (b) nodes averaged over the 7 network experiments in Table 8. Dark shades indicate high and light shades indicate low S-S (S-H) propensity.

half-cysteines perhaps enable the high prediction rate of a cysteine's disulfide bonding state. Note that in Figure 10, beyond ±5 residues from the central half-cystine (coinciding with the selected network window size) the preferences for any secondary structure are greatly reduced.

Figure 11 is a graphical depiction of the weights averaged from the seven network experiments. Note that cysteines at positions ±3 are not very conducive towards disulfide bond formation. This can be explained by the frequent occurrence of CYS-x-x-CYS in heme and metal binding proteins. However, cysteines at position ±1 increase the propensity considerably. This can be explained by the frequent occurrence of CYS-CYS in extracellular proteins, where the cysteines can form a basis for linking three chain segments in close proximity. Figure 11 also shows a positive influence of closely centered β-sheet forming residues such as ILE, TYR, and THR on disulfide bond formation.

The contribution an individual amino acid may have towards disulfide bond formation, irrespective of window position, can be seen in Figure 12.

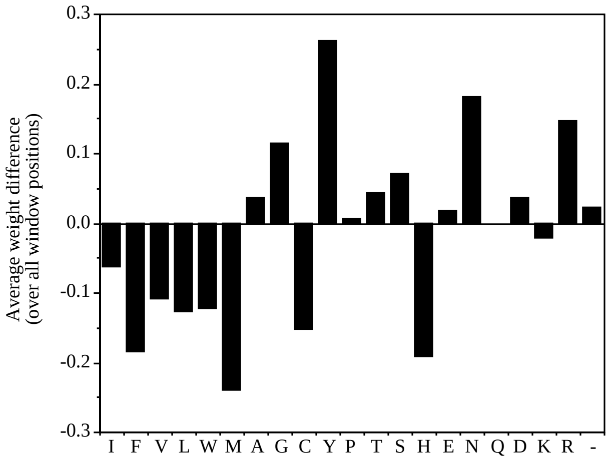


Figure 12. Amino acid contribution to disulphide bond formation. Weights from the 7 network experiments in Table 8 were averaged for each amino acid over all window positions. Bars represent the weights to the S-H node subtracted from the weights to the S-S node. Bars above the midline indicate a propensity to form S-S bonds, and those below tend to form S-H bonds.

One clear pattern is that the residues contributing *towards* S-S bond formation are polar and/or charged while those *against* formation are primarily hydrophobic. The effects of a locally hydrophobic environment could help to bury a cysteine to make it less accessible to other cysteines, thus reducing the chances of disulfide bond formation. Conversely, the effects of a locally hydrophilic environment could help to maintain cysteines in solution thus making them more accessible to one another and to increases the chances of disulfide bond formation.

The most striking features in Figure 12 exist between similar amino acids. TYR, for example, is highly conducive towards disulfide bond formation, yet PHE and TRP disfavor formation quite strongly. Electrostatic interaction between the edge of aromatic rings and sulfur atoms is found to be more frequent between aromatics and half cysteines than with aromatics and free cysteines. Figure 13 also suggests that TYR will favor disulfide bond formation over the other aromatics simply because PHE and TRP lack hydrophilic character. Likewise, ARG suggests S-S formation more strongly than LYS. Again, hydrophilic arguments find ARG more polar and thus more favorable for S-S formation. Less obvious, however, is the strong S-S propensity of ASN relative to GLN. Perhaps it is ASN's smaller size that better enables the close approach of a potential half-cystine. Consistent with this, the S-S propensity of GLY, ASP and SER exceed that of their slightly larger counter-

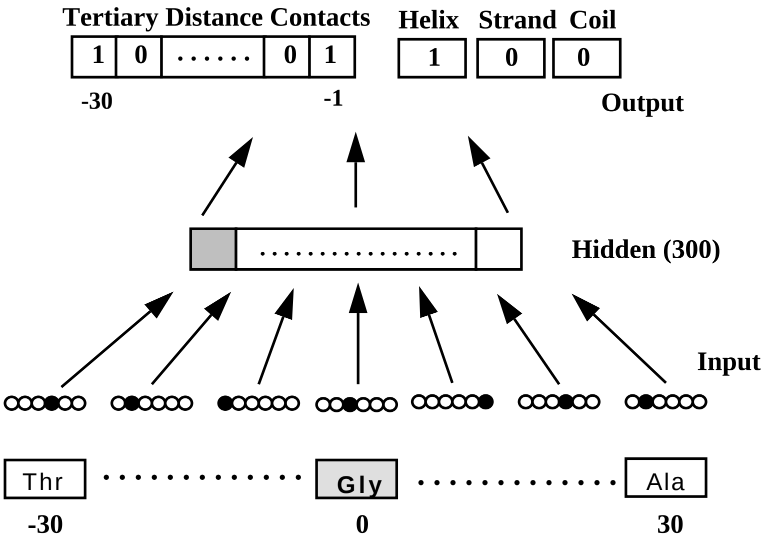


Figure 13. Network for prediction of protein tertiary structure. Input window is 30 residues to either side of the residue of interest, each represented by 20 nodes (one of which is activated). The output level consists of two parts; a window of 30 residues corresponding to those to the left of the central in the input which contains a 0 or 1 reflecting whether the residue is within 8Å of the central position. The other 3 output nodes specify the secondary structural type of the central residue.

parts ALA, GLU and THR. These differences in S-S propensity between otherwise very similar amino acids may make feasible the stabilization and/or destabilization of disulfide bonds through the site-directed mutagenesis of sequences surrounding half-cysteines.

The results of this network analysis suggest that tertiary structure features, such as disulfide bond formation, may be found in local sequence information. More experiments will need to be conducted to further exploit the information content in local amino acid sequence. Perhaps this will suggest a new twist to protein structure prediction.

6. Tertiary Structure Prediction with Neural Networks

Bohr, *et al*, [1990] recently reported the use of a feedfoward neural network trained by backpropagation on a class of functionally homologous proteins to predict the tertiary folding pattern of another member of the same functional class from sequence alone. The basis of this approach is that the commonly used binary distance matrix representation of tertiary protein structure, will be similar for members of a homologous protein family. In this representation the protein sequence is plotted along both the vertical and horizontal axes and points are placed on the graph to indicate where two C_α positions are within a specified distance in the three-dimensional structure. The network using tertiary structure information given as binary distance constraints between C_α atoms as well as a three-state model of secondary structure in the output layer and a sliding window of amino acid sequence as the input layer of a three-layer network is shown in Figure 13.

The input layer encompassed a window of -30 to +30 residues around the residue of interest (central residue) and the output a window of the 30 residues preceding the central residue. For input, each amino acid position was defined by 20 nodes each with a value of zero except for the one corresponding to the actual amino acid which had a value of one. The output layer consisted of 33 nodes, 30 representing the residues preceding the central residue and having values of zero or one depending on whether the distance to the central residue was less than or greater than 8 Å (in some cases 12 Å was used) respectively, and three nodes indicating secondary structure of helix, sheet, or coil.

This network is characterized by a very large number of computational nodes and variable weights. For input 1220 units (20x61) were used, in the hidden layer 300-400 units, and in the output 33 units. The total number of weighted links is therefore 375,900 or 501,200 for the two types of networks used. Clearly, a network containing this many weights has the capacity to memorize the small training set of 13 protease structures. The learning of the training set to a level of 99.9% on the binary distance constraints and 100% on the secondary structure assignment, indicates that the network memorizes the training set effectively, but is unlikely to incorporate generalizations. Thus, although the architecture is quite different, the application of this feedforward network is analogous to an associative memory network.

This network is quite similar to the associative memory Hamiltonian approach which has been applied for tertiary structure prediction [Friedrichs & Wolynes, 1989], thus raising the possibility that an associative memory type neural network may be useful for the storage and retrieval of protein three-dimensional folding patterns. However, it is doubtful whether this approach can predict tertiary structure of proteins which are not homologous to proteins on which the network was trained

7. Long Range Goals

While the ultimate goal of protein structural prediction is obviously to produce a complete set of three-dimensional atomic coordinates solely from the amino acid sequence, the best approach to this goal and the most important intermediate goals are still not defined. First, it should be realized that there is no such thing as a unique set of three-dimensional coordinates of a

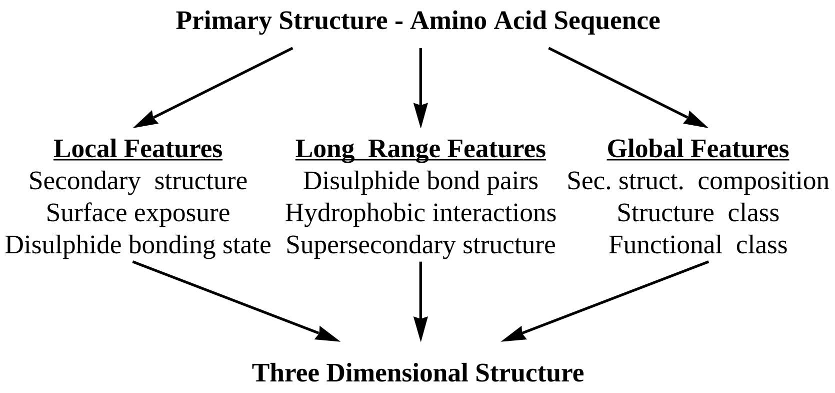


Figure 14. A possible strategy for protein structure prediction.

protein: i.e. all proteins are mobile to a greater or lesser degree and most can assume different conformations depending on environment, ligands or substrates, or complex formation. This structural variability has been observed both by NMR methods in solution and X-ray studies in crystals. The database for most theoretical studies, however, concentrates on an equilibrium or most stable conformation usually as observed in a crystal structure. Our goal, currently, must be narrowed to determining this “sample conformation” which likely corresponds to one of the minimum energy states. Now the question arises as to whether it is possible to determine this “protein structure” or at least an approximation of it from information contained in the structural and sequence databanks. It now appears that in some cases this is possible and in others the data is insufficient. For example, highly homologous proteins likely share very similar structures, while on the other hand large classes of proteins exist for which little or no structural information is available such as membrane proteins and specialized structural proteins.

Thus, a more practical if less idealistic approach, will be to concentrate efforts on the prediction of well understood structural features such as secondary structure, surface exposure, disulfide bond formation, etc. while keeping sight of the final goal of predicting a complete tertiary structure. This stairstep approach will not only provide valuable tools for molecular biologists, biochemists and protein engineers, but will also provide insight into protein structure by forcing an overall critical view of the set of known protein structures. Figure 14 illustrates the overall scheme in this approach to protein structure prediction.

8. Conclusions

The studies discussed above clearly demonstrate the power of the artificial neural network in extracting information from the protein structure database and extrapolating to make predictions of protein structural features from sequence alone. It should also be clear that so far almost all studies have utilized simple backpropagation networks. While these types of networks will continue to be widely used, it may be that the next round of advances in protein structure will involve other types of networks such as associative memory, Kohonen, or Hebbian (see, e.g., Steeg's chapter in this volume). Already, the promise of an associative memory approach has been observed. Neural networks comprise a powerful set of tools which have reached the stage where biochemists and structural biologists, and not just computer scientists, can now attack the problems of their choice. The results of these studies will depend on their ingenuity in problem formulation, network design and the informational storage of the databases. We can look forward to a rapid growth in the number of biologists using these methods.

References

Andreassen, H., Bohr, H., Bohr, J., Brunak, S., Bugge, T., Cotterill, R. M. J., Jacobsen, C., Kusk, P., Lautrup, B., Petersen, S. B., Saermark, T., & Ulrich, K. (1990). Analysis of the Secondary Structure of the Human Immunodeficiency Virus (HIV) proteins p17, gp120, and gp41 by Computer Modeling Based on Neural Network Methods. *J. Acquired Immune Deficiency Syndromes, 3*, 615-622.

Anfinsen, C. G. (1973). Principles that Hovern the Golding of Protein Vhains. *Science, 181*, 223.

Bernstein, F. C., Koetzle, T. F., Williams, G. J. B., Meyer, E. F. J., Brice, M. D., Rodgers, J. R., Kennard, O., Shimanouchi, T., & Tasumi, M. (1977). The Protein DataBank: A Computer-based Archival File for Macromolecular Structures. *J. Mol. Biol., 112*, 535-42.

Blundell, T., Sibanda, B. L., & Pearl, L. (1983). Three-dimensional Structure, Specificity and Catalytic Mechanism of Renin. *Nature, 304*, 273-275.

Bohr, H., Bohr, J., Brunak, S., Cotterill, R. M., Lautrup, B., Norskov, L., Olsen, O. H., & Petersen, S. B. (1988). Protein Secondary Structure and Homology by Neural Networks. The Alpha-helices in Rhodopsin. *Febs Lett, 241*(1-2), 223-8.

Bohr, H., Bohr, J., Brunak, S., Cotterill, R. M. J., Fredholm, H., Lautrup, B., & Petersen, S. B. (1990). A Novel Approach to Prediction of the 3-dimensional Structures of Protein Backbones by Neural Networks. *Febs. Lett., 261*, 43-46.

Chou, P. Y., & Fasman, G. D. (1974a). Conformational Parameters for Amino Acids in Helical, Sheet and Random Coil Regions From Proteins. *Biochem., 13*, 211.

Chou, P. Y., & Fasman, G. D. (1974b). Prediction of Protein Conformation. *Biochem., 13*, 222.

Covell, D. G., & Jernigan, R. L. (1990). Conformations of Folded Proteins in Restricted Spaces. *Biochem., 29*, 3287-3294.

Crick, F. (1989). The Recent Excitement About Neural Networks. *Nature, 337*, 129-132.

Eisenberg, D. (1984). Three-dimensional Structure of Membrane and Surface Proteins. *Ann. Rev. Biochem., 53*, 595-623.

Friedrichs, M. S., & Wolynes, P. G. (1989). Toward Protein Tertiary Structure Recognition By Means of Associative Memory Hamiltonians. *Science, 246*, 371-373.

Garnier, J., Osguthorpe, D. J., & Robson, B. (1978). Analysis of the Accuracy and Implications of Simple Methods for Predicting the Secondary Structure of Globular Proteins. *J. Mol. Biol., 120*, 97.

George, D. G., Barker, W. C., & Hunt, L. T. (1986). The Protein Identification Resource (PIR). *Nucl. Acids Res., 14*, 11-15.

Greer, J. (1981). Comparative Model-building of the Mammalian Aerine Proteases. *J. Mol. Biol., 153*, 1027-1042.

Holbrook, S. R., Muskal, S. M., & Kim, S. H. (1990). Predicting Surface Exposure of Amino Acids from Protein Sequence. *Protein Eng, 3*(8), 659-65.

Holley, L. H., & Karplus, M. (1989). Protein Secondary Structure Prediction with a Neural Network. *Proc Natl Acad Sci U S A, 86*(1), 152-6.

Horne, D. S. (1988). Prediction of Protein Helix Content from an Autocorrelation Analysis of Sequence Hydrophobicities. *Biopolymers, 27*(3), 451-477.

Johnson, W. C., Jr. (1990). Protein Secondary Structure and Circular Dichromism: A Practical Guide. *Proteins*, *7*, 205-214.

Kabsch, W., & Sander, C. (1983). Dictionary of Protein Secondary Structure: Pattern Recognition of Hydrogen-Bonded and Geometrical Features. *Biopolymers, 22*, 2577-2637.

Krigbaum, W. R., & Knutton, S. P. (1973). Prediction of the Amount of Secondary Structure in a Globular Protein from Its Aminoacid Composition. *Proc. Nat. Acad. Sci. USA, 70*(10), 2809-2813.

Lee, B. K., & Richards, F. M. (1971). The Interpretation of Protein Structures: Estimation of Static Accessibility. *J. Mol. Biol., 55*, 379-400.

Levin, J. M., & Garnier, J. (1988). Improvements in a Secondary Structure Prediction Method Based on a Search for Local Sequence Homologies and Its Use as a Model Building Tool. *Biochim. Biophys. Acta, 955*(3), 283-295.

Lewis, P. N., Momany, F. A., & Sheraga, H. A. (1973). Chain Reversal in Proteins. *Biochim. Biophys. Acta, 303*, 211-229.

Lim, V. I. (1974a). Algorithms for Predictions of Alpha-Helical and Beta-Structural Regions in Globular Proteins. *J. Mol. Biol., 88*, 873.

Lim, V. I. (1974b). Structural Principles of the Globular Organization of Protein Chains. A Stereochemical Theory of Globular Protein Secondary Structure. *J. Mol. Biol., 88*, 857.

Mathews, B. W. (1975). Comparison of the Predicted and Observed Secondary Structure of T4 Phage Lysozyme. *Biochim. Biophys. Acta, 405*, 442-451.

McGregor, M. J., Flores, T. P., & Sternberg, M. J. (1989). Prediction of Beta-turns in Proteins Using Neural Networks. *Protein Eng, 2*(7), 521-6.

Muskal, S. M., Holbrook, S. R., & Kim, S. H. (1990). Prediction of the Disulfide-bonding State of Cysteine in Proteins. *Protein Eng, 3*(8), 667-72.

Muskal, S. M., & Kim, S.-H. (1992). Predicting Protein Secondary Structure Content: A Tandem Neural Network Approach. *J Mol Biol, in press.*,

Nishikawa, K., Kubota, Y., & Ooi, T. (1983). Classification of Proteins into Groups Based on Amino Acid Composition and Other Characters. I. Angular Distribution. *J. Biochem., 94*, 981-995.

Nishikawa, K., & Ooi, T. (1982). Correlation of the Amino Acid Composition of a Protein to Its Structural and Biological Characteristics. *J. Biochem., 91*, 1821-1824.

Nishikawa, K., & Ooi, T. (1986). Amino Acid Sequence Homology Applied to the Prediction of Protein Secondary Structures, and Joint Prediction with Rxisting Methods. *Biochim. Biophys. Acta, 871*, 45-54.

Pascarella, S., & Bossa, F. (1989). PRONET: A Microcomputer Program for Predicting the Secondary Structure of Proteins with a Neural Network. *CABIOS, 5*, 319-320.

Qian, N., & Sejnowski, T. J. (1988). Predicting the Secondary Structure of Globular Proteins Using Neural Network Models. *J Mol Biol, 202*(4), 865-84.

Richardson, J. S. (1981). The Anatomy and Taxonomy of Protein Structure. *Adv. in Prot. Chem., 34*, 167-339.

Rose, G. D. (1978). Prediction of Xhain Rurns in Globular Proteins on a Hydrophobic Basis. *Nature* (London), *272*, 586.

Rose, G. D., Geselowitz, A. R., Lesser, G. J., Lee, R. H., & Zehfus, M. H. (1985). Hydrophobicity of Amino Acid Residues in Globular Proteins. *Science, 229*, 834-838.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning Representations by Back-propagating Errors. *Nature*, 323, 533-536.

Rumelhart, D. E., McClelland, J. L., & group, t. P. r. (1986). *Parallel Distributed Processing: Explorations in the Microstructure of Cognition* . Cambridge, MA: MIT Press.

Venkatachalam, C. M. (1968). Stereochemical Criteria for Polypeptides and Proteins. V. Conformation of a Aystem of Three Linked Peptide Units. *Biopolymers, 6*, 1425-1436.

Warme, P. K., Momany, F. A., Rumball, S. V., Tuttle, R. W., & Scheraga, H. A. (1974). Computation of Structures of Homologous Proteins. Alpha-lactalbumin from Lysozyme. *Biochem., 13*, 768-782.

Weiner, P. K., & Kollman, P. A. (1981). AMBER: Assisted Model Building with Energy Refinement. A General Program for Modeling Molecules and their Interactions. *J. Comp. Chem., 2*, 287-303.

Weiner, S. J., Kollman, P. A., Case, D. A., Singh, U. C., Ghio, C., Alagona, G., Profeta, S., Jr., & Weiner, P. (1984). A New Force Field for Molecular Mechanical Simulation of Nucleic Acids and Proteins. *J. Am. Chem. Soc., 106*, 765-784.

Wetlaufer, D. B. (1973). Nucleation, Rapid Folding and Globular Intrachain Regions in Proteins. *Proc. Natl. Acad. Sci. USA, 70*, 697.

Zvelebil, M. J., Barton, G. J., Taylor, W. R., & Sternberg, M. J. (1987). Prediction of Protein Secondary Structure and Active Sites Using the Alignment of Homologous Sequences. *J. Mol. Bio., 195*(4), 957-61.