An Analysis of Representations for Protein Structure Prediction

by

Barbara K. Moore Bryant

S.B., S.M., Massachusetts Institute of Technology (1986)

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 1994

Signature of Author .....
Department of Electrical Engineering and Computer Science
September 2, 1994

Certified by .....
Patrick H. Winston
Professor, Electrical Engineering and Computer Science
Thesis Supervisor

Certified by .....
Tomas Lozano-Perez
Professor, Electrical Engineering and Computer Science
Thesis Supervisor

Accepted by .....
Frederic R. Morgenthaler
Chairman, Departmental Committee on Graduate Students

An Analysis of Representations for Protein Structure Prediction

by

Barbara K. Moore Bryant

Submitted to the Department of Electrical Engineering and Computer Science
on September 2, 1994, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

Protein structure prediction is a grand challenge in the fields of biology and computer science. Being able to quickly determine the structure of a protein from its amino acid sequence would be extremely useful to biologists interested in elucidating the mechanisms of life and finding ways to cure disease. In spite of a wealth of knowledge about proteins and their structure, the structure prediction problem has gone unsolved in the nearly forty years since the first determination of a protein structure by X-ray crystallography.

In this thesis, I discuss issues in the representation of protein structure and sequence for algorithms which perform structure prediction. There is a tradeoff between the complexity of the representation and the accuracy to which we can determine the empirical parameters of the prediction algorithms. I am concerned here with methodologies to help determine how to make these tradeoffs.

In the course of my exploration of several particular representation schemes, I find that there is a very strong correlation between amino acid type and the degree to which residues are exposed to the solvent that surrounds the protein. In addition to confirming current models of protein folding, this results suggests that solvent exposure should be an element of protein structure representation.

Thesis Supervisor: Patrick H. Winston
Title: Professor, Electrical Engineering and Computer Science

Thesis Supervisor: Tomas Lozano-Perez
Title: Professor, Electrical Engineering and Computer Science

Acknowledgments

For their support, I thank my thesis committee, the members of the Artificial Intelligence Laboratory, Jonathan King's group, the biology reading group, Temple Smith's core modeling group at Boston University, NIH, my friends and family.

The following people are some of those who have provided support and encouragement in many ways. To those who I've forgotten to list, my apologies, and thanks.

Bill Bryant, Ljubomir Buturovic, John Canny, Peggy Carney, Kevin Cunningham, Sudeshna Das, Cass Downing-Bryant, Rich Ferrante, David Gifford, Lydia Gregoret, Eric Grimson, Nomi Harris, Roger Kautz, Jonathan King, Ross King, Tom Knight, Kimberle Koile, Rick Lathrop, Tomás Lozano-Pérez, Eric Martin, Mark Matthews, Michael de la Maza, Bettina McGimsey, Scott Minneman, Doug Moore, Jan Moore, Stan Moore, Ilya Muchnik, Raman Nambudripad, Sundar Narasimhan, Carl Pabo, Marilyn Pierce, Geoff Pingree, Tomaso Poggio, Ken Rice, Karen Sarachik, Richard Shapiro, Laurel Simmons, Temple Smith, D. Srikar Rao, Susan Symington, James Tate, Bruce Tidor, Bruce Tidor, Lisa Tucker-Kellogg, Paul Viola, Bruce Walton, Teresa Webster, Jeremy Wertheimer, Jim White, Patrick Winston, and Tau-Mu Yi.

Contents

List of Figures

List of Tables

n

Chapter 1

Overview

1.1 Why predict protein structures?

Proteins play a key role in innumerable biological processes, providing enzymatic action, cell and extracellular structure, signalling mechanisms, and defense against disease [Stryer, 1988]. Much research in the pharmaceutical industry is geared toward understanding biological processes and designing new proteins or molecules that interact with proteins. Because a protein's interactions with other molecules are governed by its three-dimensional structure, a central problem in this research is determining the three-dimensional structure of proteins.

Most known protein structures were determined by interpreting the X-ray diffraction patterns from protein crystals. Protein purification and crystallization is extremely difficult, and interpreting the X-ray diffraction patterns is not straightforward. Some small protein structures can be solved using nuclear magnetic resonance techniques on proteins in solution, but this is not yet possible for most proteins. Currently we know the shape of hundreds of proteins, but there are some hundreds of thousands of proteins of interest.

We do know the molecular formulae and covalent bonding structure of the proteins. Proteins are composed of smaller molecules, called amino acids. The structure of an amino acid is illustrated in Figure 1-1. The molecule is arranged in a tetrahedral geometry around the central carbon, called the alpha carbon. Amino acids differ

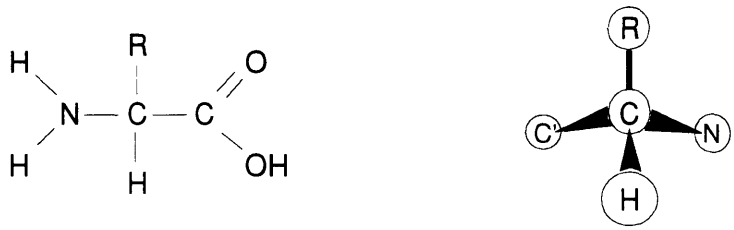


Figure 1-1: A generic amino acid. The "R" represents a variable side-chain.

from each other in the side chain, represented by "R" in the figure. There are 20 different types of amino acids in proteins (Figure 1-2). The amino acids vary in size, polarity (whether they are charged or not), hydrophobicity (whether they "fear water"), and other chemical properties.

The amino acids are covalently bonded together (Figure 1-3) to form a long chain of amino acid residues (Figure 1-4). Typically there are hundreds of amino acid residues in a protein.

The backbone of the amino acid residue chain has three atoms (N—C—C) from each residue, and therefore three bonds per residue. Two of these bonds allow fairly free rotation (Figure 1-5). The protein can therefore potentially take on an enormous number of different shapes, or conformations. There is additional conformation freedom in most of the sidechains.

For the past 40 years, researchers have been inventing methods for predicting a protein's three-dimensional structure from its amino acid sequence [Fasman, 1989]. Many people have analyzed the protein structures that are known, hoping to uncover useful principles for the prediction problem.

So far, protein structure prediction methods have met with limited success and it is clear that much more needs to be learned before we will be able to reliably determine a protein's shape without the aid of the X-ray crystallographer.

This thesis investigates how to represent proteins on the computer. The representation that we choose shapes the information that we gather and use in modeling and predicting. For the various kinds of information that we might want to represent about a protein, there are a number of questions we might ask. How redundant are

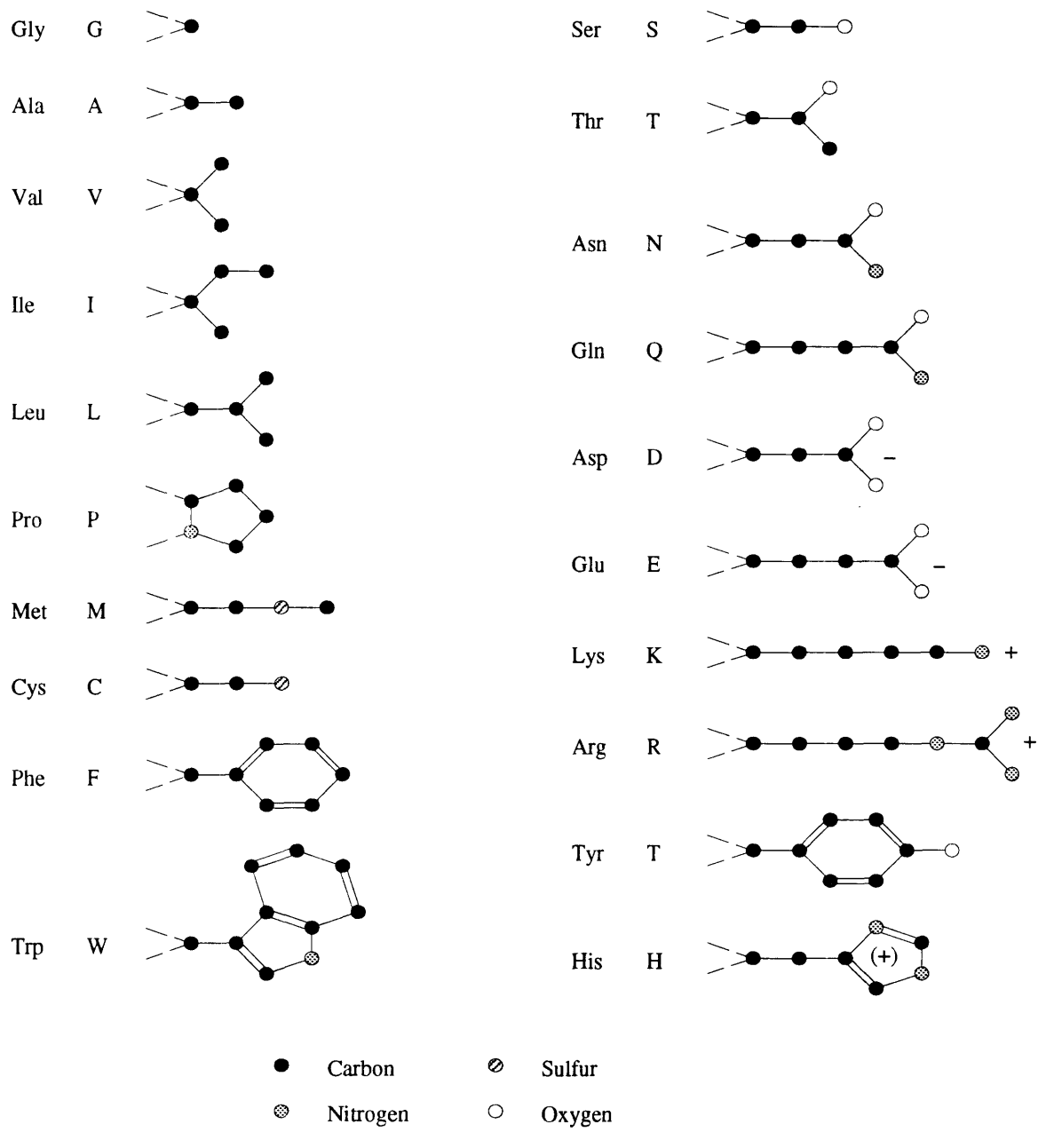


Figure 1-2: The sidechains of the 20 types of amino acids found in proteins. Hydrogen atoms are not shown. Dashed lines indicate backbone bonds. The N in the proline (Pro) residue is the backbone nitrogen atom. For each amino acid residue, the three-letter and one-letter code are given.

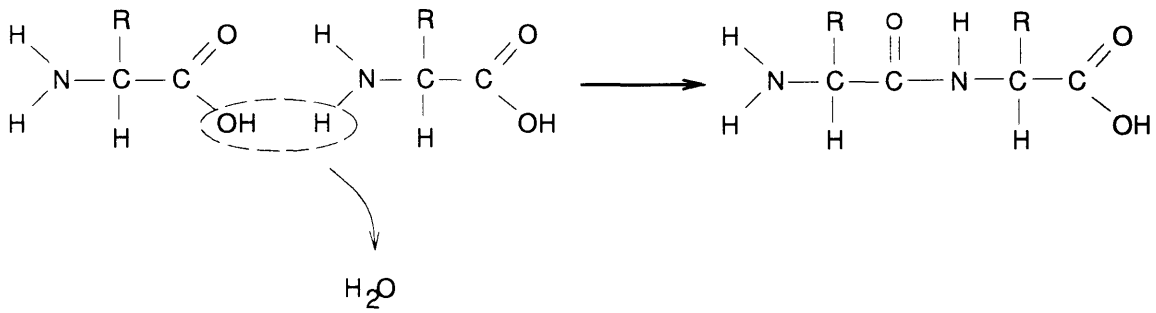


Figure 1-3: Covalent linking of two amino acids to form a dipeptide.

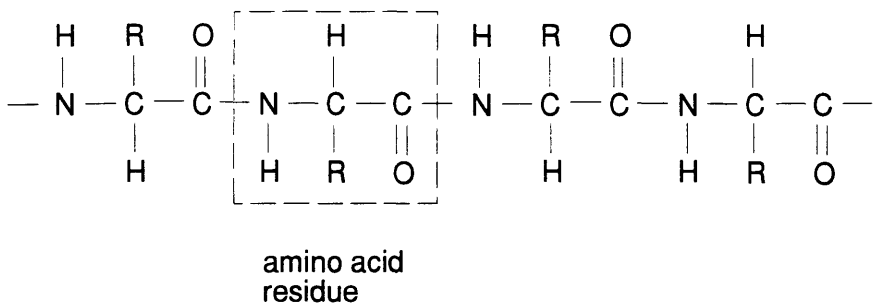


Figure 1-4: A protein is a chain of amino acid residues.

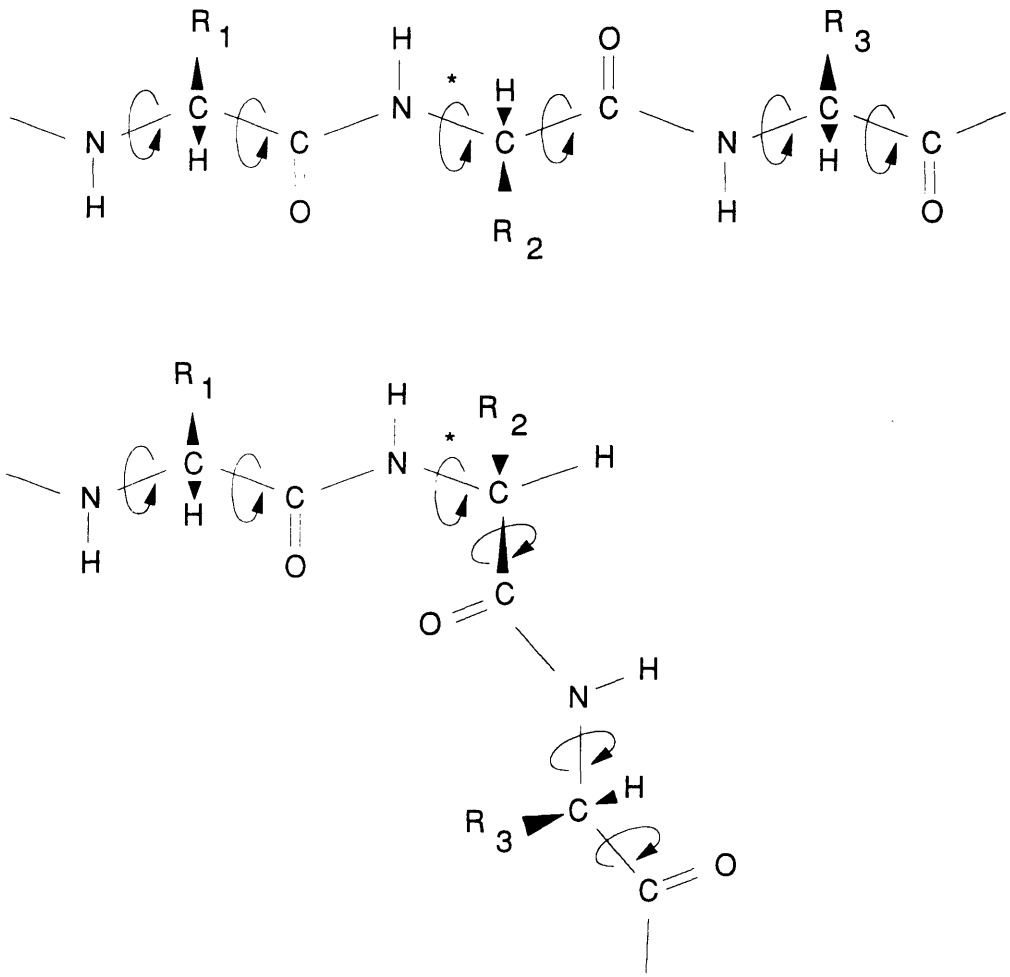


Figure 1-5: A protein is flexible. The backbone of a protein contains three bonds per residue, two of which allow free rotation as indicated by the arrows in the diagram. The lower conformation pictured is a conformation obtained from the upper conformation by rotating 180° about the starred (*) backbone bond.

the various types of information? How do we use the representations? How do we combine different types of information? How do we choose which representation we want?

The rest of this chapter provides more background about the protein prediction problem, and summarizes my approach and results.

1.2 Background

Amino acid sequences of proteins were determined years before any protein structures were known. There was a lot of interest in the structures because it was known that the shapes of proteins determine how they interact with other molecules and therefore how they function in a cell.

The three-dimensional structure of individual amino acids and dipeptides (bonded pairs of amino acids) had been determined by analyzing diffraction patterns of X-rays through crystals. People expected that the amino acid residues in large protein molecules would fit together in neat, regular, repeating patterns. In 1951, seven years before the first protein structure was observed, two protein backbone conformations were predicted based on the known structure of amino acids [Pauling and Corey, 1951]. The patterns were a helix shape and an extended "strand" shape that could pack next to other strands in sheets.

The first protein structure, myoglobin, was determined by means of X-ray crystallography [Kendrew and others, 1958]; see Figure 1-6, which was produced by the Molscript program [Kraulis, 1991]. People were dismayed at the apparent spatial disorder within each protein molecule. The protein shapes were a far cry from the regular repetitive packing that had been imagined.

On the other hand, there were a few encouraging aspects of protein structure. First of all, the predicted helix and strand structures did occur quite often in the proteins. Figure 1-7 shows a diagram of the backbone of ribonuclease A, containing both helices and sheets. The way the individual helices and strands packed together was not planar or regular, and much of the protein molecule looped around in very

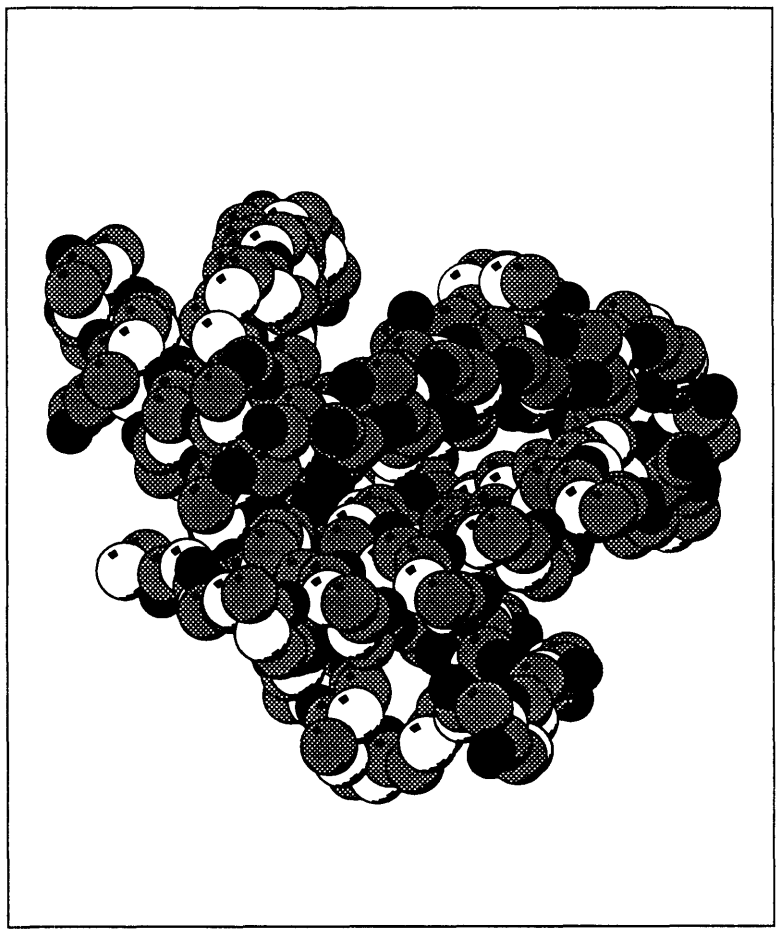


Figure 1-6: The first observed protein structure. Myoglobin has 153 residues. Only the non-hydrogen backbone atoms are shown. Black spheres represent oxygen atoms; grey spheres represent carbons; white spheres represent nitrogens. Drawn by the Molscript program.

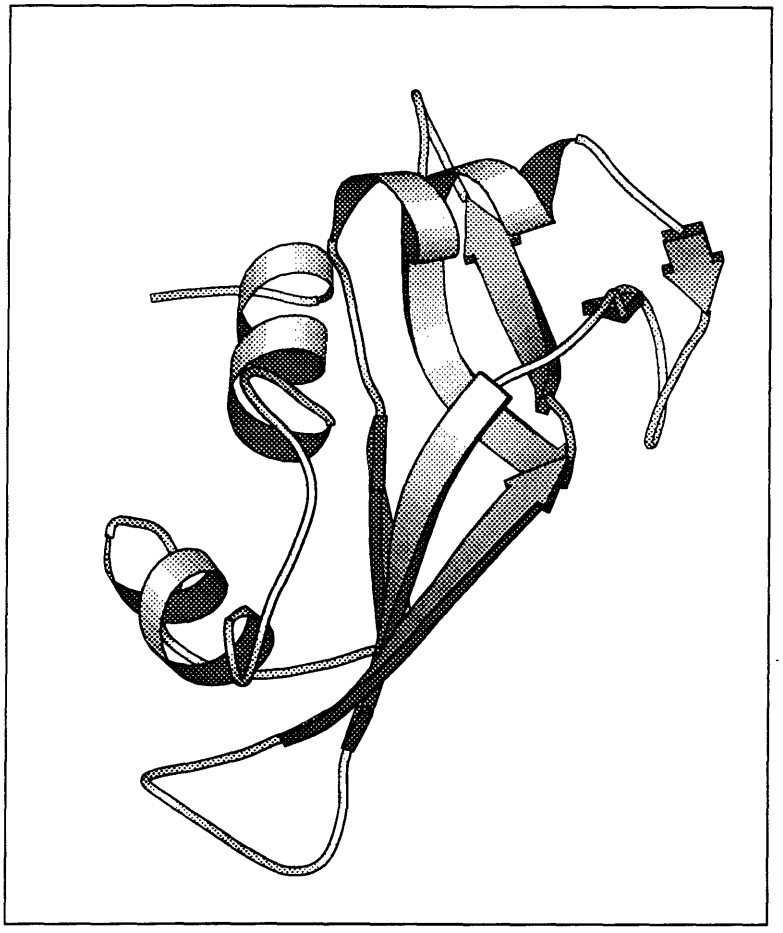


Figure 1-7: Structure of Ribonuclease A. The backbone of the protein is shown, with alpha (helices) and strand (arrows) regions shown. Drawn by the Molscript program.

irregular shapes. The helices and strands themselves were often twisted, bent or kinked.

In spite of the seeming irregularity of the protein structures, a given sequence of amino acid residues always seemed to fold to the same complex structure. It was shown that ribonuclease and other proteins could be denatured and refolded to their native structure [Anfinsen *et al.*, 1961]. This fact was tantalizing. It suggested that there must be some way to model the forces at work on and in the protein, such that one could predict the protein's three-dimensional structure from its amino acid sequence.

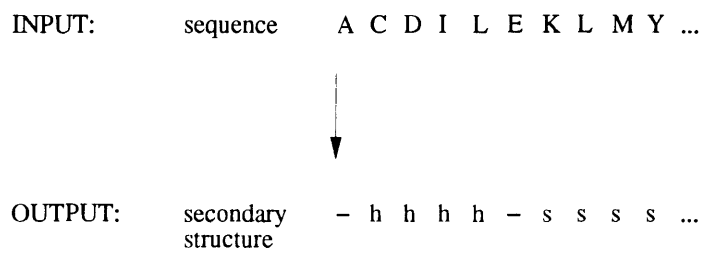


Figure 1-8: The secondary-structure prediction problem. Secondary structure labels are "h" (helix), "s" (strand), and "-" (other).

1.2.1 Secondary structure prediction

The fascination with helices and strands, the local structure found throughout proteins, has continued unabated. A hierarchy of protein structure description was defined. The first level, primary structure, is defined as the sequence of amino acid residues in the protein. Secondary structure is defined to be the local protein structure, for example, strands and helices. Tertiary structure is the conformation of one protein: the three-dimensional positions of all the protein's atoms.

Much effort has been focussed on the prediction of secondary structure (Figure 1-8). In this problem, the goal is to find the type of secondary structure in which each residue in the protein occurs. The input is the amino acid sequence of the protein.

Early models of protein folding were based on the idea of secondary structure nucleation followed by propagation in a zipper-like effect along the protein chain [Zimm and Bragg, 1959, Lifson and Roig, 1961]. These models were used to interpret real folding data on polypeptides [Scheraga, 1978]. The Chou-Fasman model is a good example of a prediction algorithm based on the nucleation-propagation folding model [Chou and Fasman, 1978]. The basic idea was that short stretches of residues in the protein might strongly favor one of the secondary structures, and act as nucleation sites for those structures. The structures would then be extended along the chain until they ran into other amino acid residues which were strongly disfavorable to that type of structure. The extent to which an amino acid "favored" a particular type of secondary structure was determined by counting the number of occurrences of each amino acid in each type of secondary structure in a set of known-structure

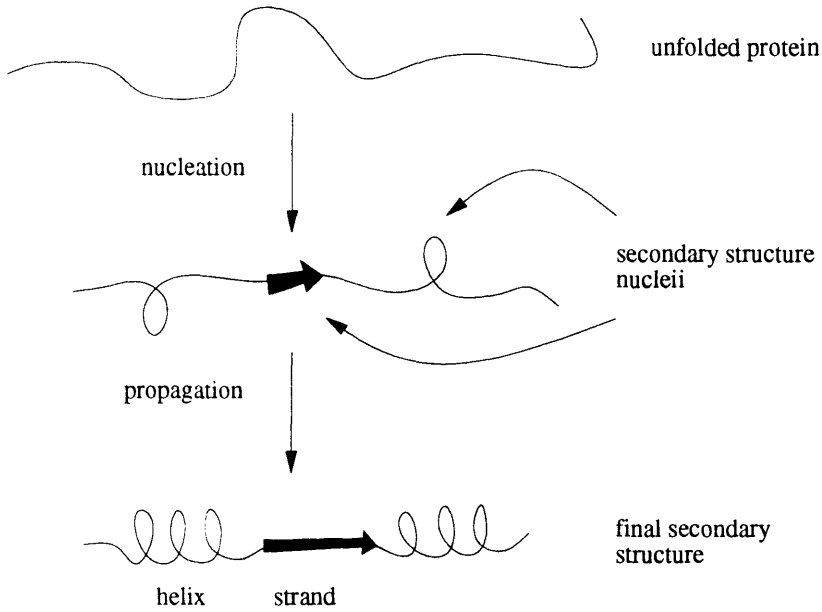


Figure 1-9: The secondary structure nucleation-propagation model heavily influenced early protein structure prediction methods.

proteins.

Many different secondary-structure methods have been tried. Most of these approaches are statistically based, relying on numbers culled from the database of known protein structures. Algorithms were developed based on information theory [Garnier *et al.*, 1978, Gibrat *et al.*, 1987], neural networksHolley89,Qian88,Stolorz91,Bohr88, pattern-matching [Cohen *et al.*, 1986], machine learning [King, 1988], and Markov random fields [Collin Stultz and Smith, 1993]. Variations were tried in the definitions of secondary structure, in the input sequence representations, and in the types of additional information provided to the algorithm [Kneller *et al.*, 1990, McGregor *et al.*, 1989, Levin *et al.*, 1993, Niermann and Kirschner, 1990, Rost and Sander, 1993a].

People observed that there seemed to be an upper barrier to the accuracy with which secondary structure could be predicted from sequence information (about 65% residues correctly predicted for three secondary structure states: helix, strand, and other). There are several possible explanations for this limit. There might not be enough structure data yet to accurately determine the empirical parameters used in the predictions [Rooman and Wodak, 1988]. There might be sufficient information but the models themselves are faulty. Or it might be that secondary struc-

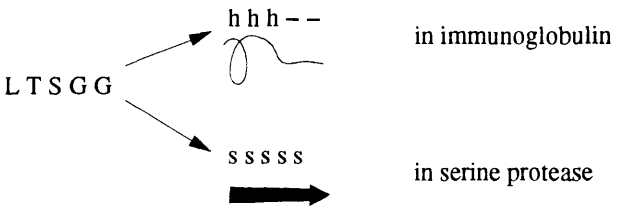


Figure 1-10: The same local sequence folds to different secondary structures in different proteins, from Argos, 1987.

ture is not determined by secondary sequence alone. Clever computational experiments were performed to try to distinguish between these possibilities. Examples were found of short amino acid sequences that had different secondary structures in different proteins (Figure 1-10) [Kabsch and Sander, 1984, Argos, 1987, Sternberg and Islam, 1990]. Most of these results pointed toward the explanation that secondary sequence information is not sufficient to uniquely determine secondary structure. The conclusion was that tertiary structure interactions between amino acid residues very far apart in sequence but close in space must be crucial in determining secondary structure.

1.2.2 Tertiary structure prediction

Even if we discovered a way to determine secondary structure perfectly, or were told the answer by an oracle, we would not be done. We want to know the overall structure of the protein, not just the secondary structure.

One strategy is to start from the results of secondary structure prediction. People have worked on the problem of packing together strands and helices into a full three-dimensional protein structure [Cohen *et al.*, 1979, Cohen and Kuntz, 1987, Hayes-Roth and others, 1986]. This approach has the potential for dealing with the problem of ambiguous or inaccurate secondary-structure prediction, by following multiple hypotheses, or by providing feedback to the secondary-structure predictor from the packing algorithm.

Another model of protein folding is strongly related to this approach of packing secondary structure pieces. The notion of building up a protein structure step by

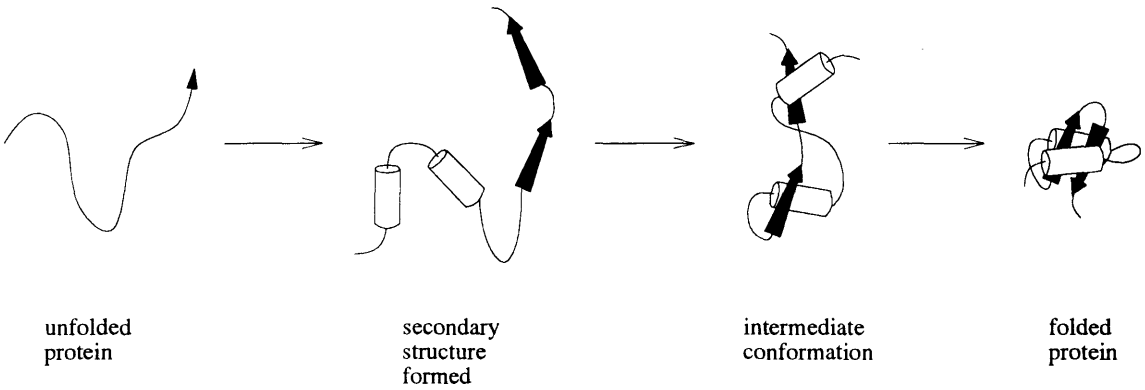


Figure 1-11: The pathway model of protein folding.

step was bolstered by the prevalent notion in the biochemical literature of folding intermediates. Levinthal [Levinthal, 1968] phrased the following argument: the conformational space of a protein is too vast to be searched in a reasonable time by a protein. A possible conclusion is that there must be some specific folding pathway, with one or a few intermediate conformations that every folding protein visits (Figure 1-11). Experimentalists found ways to trap and characterize protein species (or collections of conformations) that they described as intermediate conformations on the folding pathway [Creighton, 1978, Bycroft *et al.*, 1990, Matouschek *et al.*, 1990, Nall, 1986, Weissman and Kim, 1991]. The partially-packed protein structures in the secondary-structure-packing prediction methods are reminiscent of this idea of folding intermediates.

We probably understand enough about atoms and molecules to correctly model, with quantum mechanics, the structure of a protein. To determine the structure, you would solve the wave equation for the entire molecule and use a partition function to find low-energy solutions for atomic positions. Unfortunately, for proteins this calculation is not practical analytically, and prohibitively expensive computationally.

People have developed approximate energy functions for proteins that estimate the molecule's potential energy as a function of position [Karplus and Petsko, 1990]. It has been postulated that the folded protein is at a global or local minimum of these energy functions. The model incorporates interactions between atoms in the molecule, and biases bond lengths and angles toward a few favored positions. These

```
sequence 1    Q R E T - - F N S I Q L E V - - N T ...
sequence 2    Q - D T P N H N S V - L D I M H R S ...
```

Figure 1-12: Two protein sequences are aligned to maximize the similarity between aligned amino acid residues.

energy functions are used to help determine protein structures from X-ray diffraction data. The energy function can be used to model the motion of a protein in time, but the model is too complex to allow simulation of motion for the time that it would take the protein to fold.

People tried to simplify the energy function and protein structure representation in order to allow simulations which would model a long enough period of time to simulate the folding of a protein [Levitt and Warshel, 1975, Godzik *et al.*, 1992, Hagler and Honig, 1978, Kuntz *et al.*, 1976, Skolnick and Kolinski, 1990]. Instead of modeling all the atoms in the molecule, amino acid residues (which have ten or twenty atoms) are represented by one or two super-atoms. Forces between atoms are replaced by mean forces between amino acid residues. The motion of the molecules is restricted in various ways to simplify computation. For example, in some schemes only movements on a lattice are allowed.

One problem with these simplifications is that it is difficult to determine whether the model has been simplified to the point of losing important information. Simplified protein dynamics is not currently a viable means of predicting protein structure.

Another approach is to align the protein sequence to the similar sequence of a known-structure protein, if one exists (Figure 1-12). A model for the protein is then built based on the known structure of the other protein. This approach is known as homology modeling, and is currently the most successful protein structure prediction method [Lee and Subbiah, 1991].

In the 1980s the "hydrophobic collapse" theory of protein folding gained favor in the field. According to this theory, a protein folds in two phases. In the first phase, the unfolded protein chain collapses quickly to a fairly compact shape, called a molten globule, and this phase is driven by the tendency for hydrophobic ("water-fearing") amino acids to avoid water and clump together. The molten globule contains some

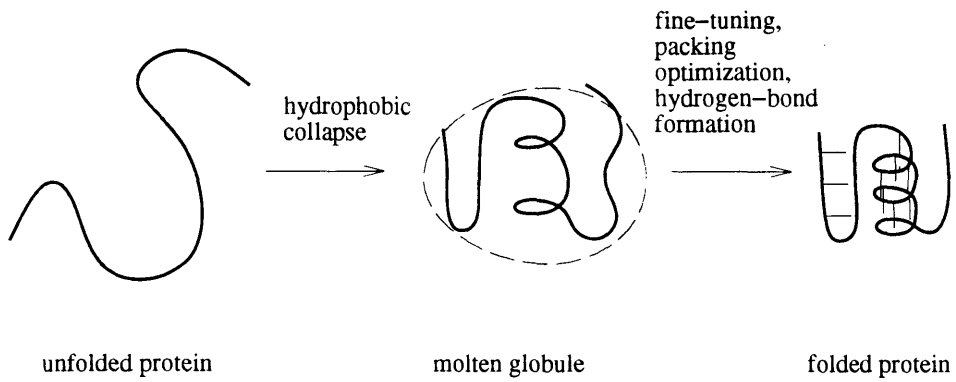


Figure 1-13: The hydrophobic collapse model of protein folding proceeds in two stages.

secondary structure (strands and helices) and has roughly the fold of the final structure, but is larger. In the second phase of folding, fine tuning occurs as the protein chain adjusts to optimize interactions between atoms. The result is a tightly packed structure, characteristic of observed folded proteins. There is now much experimental support for the hydrophobic collapse folding theory.

At this same time, people began looking for ways to judge the quality of a proposed protein structure. These methods were strongly influenced by the hydrophobic collapse folding model. A well-folded protein was modeled to have charged and polar (partially charged) parts of the protein on the outside, and hydrophobic parts on the inside (Figure 1-14). "Pseudopotential" functions were developed to incorporate this idea. These functions are similar to the simplified energy functions used in protein folding simulations. Experiments were done showing that pseudopotential functions could discriminate between correct and incorrect structures for one sequence [Baumann *et al.*, 1989, Chiche *et al.*, 1990, Holm and Sander, 1992, Vila *et al.*, 1991].

This discrimination by means of a pseudopotential function between correctly folded and misfolded proteins led naturally to an extension of homology modeling in which sequences were compared directly to a set of candidate structures [Wodak and Rooman, 1993, Blundell and Johnson, 1993, Fetrow and Bryant, 1993]. First the sequence is aligned onto each candidate structure, then the pseudopotential is used to determine which sequence-structure alignment is the correct one (Figure 1-15). This

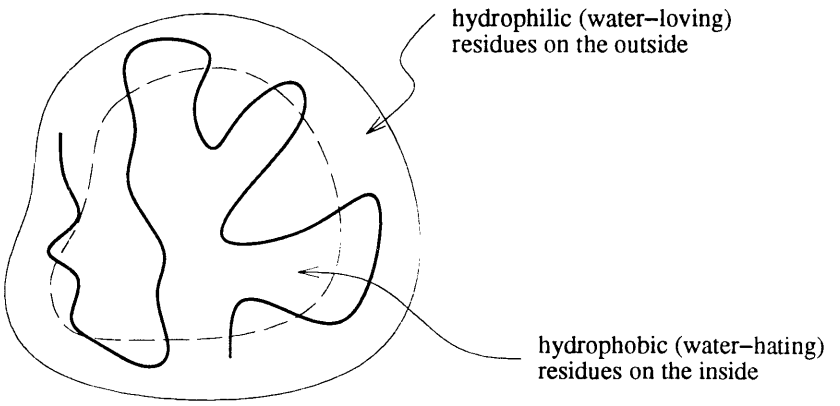


Figure 1-14: Sketch of a correctly folded protein.

approach is currently very popular. It is sometimes referred to as *inverse folding*: instead of predicting structure from sequence, we have a pseudopotential function to evaluate how likely the sequence is to have been "generated" by a structure. The sequence-structure alignment is called *threading* because the sequence is threaded onto (aligned with) each structure. Arguments have been made, based on the fraction of new additions to the structure database which represent truly new folds, to the effect that we have seen a large fraction of all protein structures, and therefore the threading prediction method is likely to succeed in a large number of cases. The structure database might also be expanded by constructing hypothetical structures out of pieces of known structures.

Many threading pseudopotentials have been formulated. Pseudopotential functions were originally based on the hydrophobic collapse folding model. A numerical value was assigned to each amino acid type to represent its hydrophobicity; this number was based on chemical experiments with a single amino acid type, or on statistical analyses of the known-structure database. In addition, each residue position in the structure has a numerical degree of exposure to the solvent. The pseudopotential functions compared the exposure of a residue's position to the residue's hydrophobicity, and assigned a low energy to buried hydrophobic residues and exposed hydrophilic residues.

Once the idea of a pseudopotential function for evaluating sequence-structure

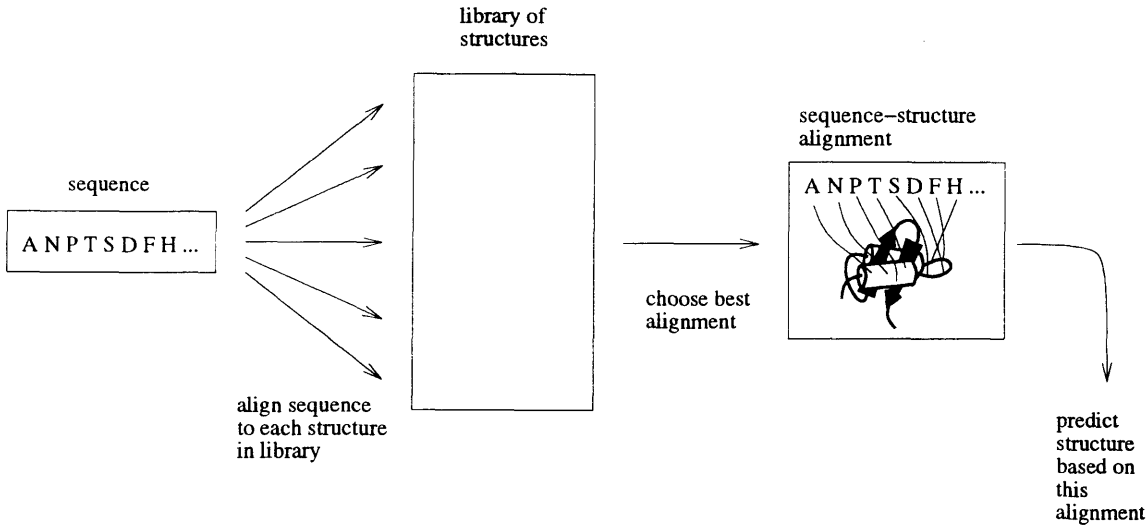


Figure 1-15: The threading method of tertiary structure prediction.

alignments was established, people experimented with incorporating other types of information in the pseudopotentials. These included:

- A residue's propensity to be in each type of secondary structure type (for example, does the residue "prefer" helix secondary structure over strand secondary structure?). This harks back to the early focus on secondary structure prediction.

- A residue's interaction with neighboring residues in the structure. Not only sequence-local neighbors, but neighbors distant in sequence but close in space could now be modeled. This was an exciting development, because it would seem to provide a way around the limitation imposed by considering only local sequence information in structure prediction.

Research on the inverse folding approach to structure prediction is actively being pursued at present. It is too early to determine the success of this method.

1.3 My work

The focus of my work is the question, "What makes for a good representation of protein sequence and structure for structure prediction algorithms?" In carrying out

this research, I am particularly interested in distinguishing the underlying models of protein folding that have influenced the structure prediction work. My approach is to compare different possible components of a protein representation. I consider a particular type of representation, in which each residue position in the database of known structures is labeled with a set of attributes. These attributes might include, for example, secondary structure type, solvent exposure, or amino acid. I use a statistical technique called contingency table analysis that allows one to tease out the relative importance of, and interaction between, the different types of information in a representation of events. In my work, the events of interest are the residue positions in the structure database. I discuss implications of this analysis for protein structure prediction. I also consider the power of the protein representations in the contexts in which they will be used, by comparing their performances in secondary and tertiary structure prediction algorithms.

In the next sections, I highlight a few of the questions I investigated.

1.3.1 Hydrophobic collapse vs. structure nucleation and propagation

My results show that a structure prediction algorithm based on the hydrophobic collapse model of protein folding should perform better than one based on secondary structure nucleation and propagation. In particular, I observe that amino acid type is more strongly correlated with solvent exposure than with secondary structure. This finding agrees with the currently popular hydrophobic collapse model of protein folding: that the first and strongest effect is the burial of hydrophobic residues in the core of the structure. Thus, my results indicate that protein structure prediction representations should include an effective model of hydrophobicity and solvent exposure. For example, it might be useful to predict solvent exposure along the protein chain instead of the commonly used secondary structure as the first step in a two-step tertiary structure prediction method.

However, I do find that where it is possible to incorporate both exposure and

secondary structure propensities, it is a good idea to do so; secondary structure propensities do add useful information.

1.3.2 Modeling hydrophobic collapse

The hydrophobic effect has been modeled in various ways. What is really going on physically is complicated. Polar solvent molecules around exposed hydrophobic residues in the protein lose hydrogen-bonding partners. On the other hand, buried polar residues may miss chances to make good hydrogen bonds. One model of the hydrophobic effect creates a fictitious hydrophobic force, in which hydrophobic atoms or residues attract one another. This pairwise interaction model is used by Casari and Sippl, for example [Casari and Sippl, 1992]. An alternative approach (for example, Bowie and colleagues [Bowie *et al.*, 1991]) looks at each residue in isolation as a singleton term in the pseudopotential and asks how hydrophobic it is and how much water it sees. If the residue is buried and hydrophobic, or exposed and hydrophilic, then the residue is happy. Which approach is better, pairwise attractive force or singleton solvent exposure?

I compare the pairwise model to the singleton model of hydrophobic collapse. The former examines the correlation between the hydrophobicities of neighboring amino acids; the latter examines the correlation between an amino acid's hydrophobicity and its solvent exposure. My analysis shows that looking at the association between amino acid type and solvent exposure at a single site is more informative than looking at the association between pairs of amino acids. The implication is that it is a good idea to model the hydrophobic effect as a first-order correspondence between amino acid hydrophobicity and solvent exposure, as opposed to as the second-order effect which is the pairing of similar types of amino acids. This turns out to be a very useful result for threading algorithms, because threading with first-order effects can be done quickly, while threading with pairwise effects is computationally expensive [Lathrop, 1994].

1.3.3 Pairwise interactions: great expectations

Many designers of pseudopotential functions assume that modeling of pairwise, triplet, and even higher-order interactions between residue positions in protein structures is necessary to distinguish misfolded proteins from correctly folded proteins. Threading methods incorporating high-order residue interactions have been developed based on this assumption.

Statistical arguments have been made about the relative importance of the higher-order terms to the singleton terms, and it was shown that in theory the pairwise terms should provide half again as much information as the singleton terms [Bryant and Lawrence, 1993]. What happened in practice? While little has been published comparing the performance of singleton to higher-order pseudopotentials, preliminary results indicate that pairwise terms do not improve the threading results. Why would this be? Perhaps the models are inadequate, or the pseudopotentials are incorrect in some way.

The analysis that I performed shed some light on the question of the importance of pairwise interactions in threading.

First of all, my statistical analysis at first glance suggests that pairwise terms should improve the threading results. They contain information not available in the single-residue attributes. In fact, by one way of measuring, the pairwise terms should contain half again as much information as the singleton terms.

However, a closer examination of the statistics shows the pairwise terms in the pseudopotential scoring functions are plagued by severe problems with low sample size. The pairwise terms do contain some additional useful information, but with low sample sizes it is swamped by noise.

To compensate for the noise, there are several things that can be done. One approach is to reduce the complexity of the data representation. I grouped the 20 amino acids into three groups based on their hydrophobicity. When I do this I have adequate sample sizes, but the singleton terms are now far more important than the pairwise terms. This could be due to the fact that the representation is too coarse, or it might be a more accurate representation of the true relative importance of pairwise

and singleton terms, or some combination of the two.

I also find that some information is accounted for by the preference of an amino acid for its neighbor's solvent exposure. This effect is not modeled by current pseudopotential functions that model pairwise interactions.

1.3.4 Amphipathicity

In looking at pairwise occurrences of amino acids, I discovered an unexpected correlation between hydrophobicity types on opposite, non-contacting sides of buried beta sheets. This might represent a sequence signal for beta strand conformation. Regardless of the reason, this, along with the other results about the importance of solvent exposure, suggested to me that I might try to incorporate some sort of amphipathicity constraint in structure prediction algorithms. I found that providing amphipathicity information to a neural net that predicts secondary structure improves its performance by a small but significant amount.

1.4 Outline of thesis

Chapter 2 describes the protein representations I use in the thesis, and gives some background on the analysis of contingency tables using loglinear models. The next four chapters (3–6) apply contingency table analysis to single-residue properties, paired residues, single residue in parallel and antiparallel beta sheets, and pairs of residues in beta sheets. The next two chapters evaluate protein representations by using them in programs that operate on proteins. In Chapter 7, I use various sequence representations as input to a neural network that predicts secondary structure. In Chapter 8, I use various structure representations in the threading program to align a structure to a sequence. In Chapters 9 and 10, I describe work in progress and summarize my conclusions. Appendix A describes some related work in protein statistics and threading.

Chapter 2

Background

In this chapter, I discuss the protein representations employed in this thesis. Then I introduce contingency table analysis, the statistical method that I use extensively in Chapters 3 through 6, and which is closely related to the pseudopotential functions I test in Chapter 8.

2.1 Knowledge representation

In this section, I discuss the particular types of knowledge representation that I consider for protein sequences and structures. I consider a residue to be the basic unit of protein structure. I represent a protein as a string of residues. Each residue has a set of attributes. In addition, the protein representation may include a list of pairs of residues that are related in some way, and each residue pair may have attributes above and beyond the individual residues' attributes. I refer to the attributes of a single residue as "singleton" attributes; those of a residue pair are "pairwise" attributes.

Each attribute can take on one of a finite set of values. A given residue is categorized by one and only one value for each of its attributes. Thus the attribute values are complete and non-overlapping. On occasion I compare or use attributes that are generalizations or specializations of each other. An attribute A_1 is a generalization

of an attribute A_2 if for any two residues r_i and r_j,

$$(A_2(r_i) = A_2(r_j)) \Longrightarrow (A_1(r_i) = A_1(r_j)),$$

where $A_1(r_i)$ is the value of attribute A_1 for residue r_i (and so on). In other words, if A_2 classifies two residues as having the same attribute values, then A_1 must also.

2.1.1 Single residue attributes

The attributes of a residue that I investigate include solvent exposure, secondary structure, and sequence.

Solvent exposure

Solvent exposure specifies the amount of a residue's surface area that is exposed to the solvent on the outside of the protein. In this thesis, the possible solvent exposure values are {**buried, exposed**}.

The solvent exposure is computed by the DSSP program using geodesic sphere integration [Kabsch and Sander, 1983]. Points on the surface of a sphere of radius equal to the sum of the radii of an atom and a water molecule are considered exposed if the water sphere centered there does not intersect with any other protein atom. The total area of these points for a residue are computed by summing over a polyhedron made of approximately equal triangles. The atomic radii are taken to be 1.40 for O, 1.65 for N, 1.87 for C_α, 1.76 for the carbonyl C, 1.8 for all side-chain atoms, and 1.4 for a water molecule. The number reported by the DSSP program is the average number of molecules in contact with each residue, which is estimated from the surface area by dividing by 10 square Angstroms.

I compute relative solvent exposure by dividing the DSSP solvent exposure by the maximum exposure for each residue, as recorded in Section B.3.

I apply a threshold of 20% to the relative solvent exposure to determine the buried and exposed labels.

Secondary structure

I use the DSSP definitions of secondary structure types **alpha helix** and **beta strand** [Kabsch and Sander, 1983]. All other residues are labeled **coil**. In some of the analysis, I further divide the beta strand category into **parallel** and **antiparallel**; this is a specialization of the {alpha, beta, coil} attribute.

The DSSP program finds alpha helix and beta strand by first determining the locations of backbone hydrogen bonds. Hydrogen bonds are determined by placing partial charges on the C (+.42e), O (-.42e), N (-.2e), and H (+.2e) atoms. The electrostatic interaction energy is calculated as

$$E = f(.42e)(.2e)\left[\frac{1}{r_{ON}} + \frac{1}{r_{CH}} - \frac{1}{r_{OH}} - \frac{1}{r_{CN}}\right],$$

where f is a dimensional factor to translate from electron units to kcals, $f = 332$, and E is in kcal/mol. e is the unit electron charge. The distance between atoms of type i and j, r_{ij}, is in angstroms. A hydrogen bond is said to exist if E is less than the cutoff -0.5 kcal/mol.

DSSP defines a "bridge" as existing between two nonoverlapping stretches of three residues each if there are two hydrogen bonds characteristic of beta structure (Figure 2-1). A "ladder" is a set of one or more consecutive bridges of identical type, and a "sheet" is a set of one or more ladders connected by shared residues. A parallel bridge aligns bridge residues in pairs as $(i-1, j-1)$, (i, j), and $(i+1, j+1)$. An antiparallel bridge aligns pairs $(i-1, j+1)$, (i, j), and $(i+1, j-1)$.

Sequence

I use several representations for sequence. The simplest is the amino acid, which has 20 different values, one for each type of amino acid found in proteins. I also (following Lifson and Sander [Lifson and Sander, 1980]) group the amino acids by hydrophobicity, obtaining three classes **hydrophobic**, **neutral**, and **polar**. Lifson and Sander use two other groupings that I employ in Chapter 6 on pairwise interactions in beta sheets. These grouped representations are generalizations of the 20-valued amino acid

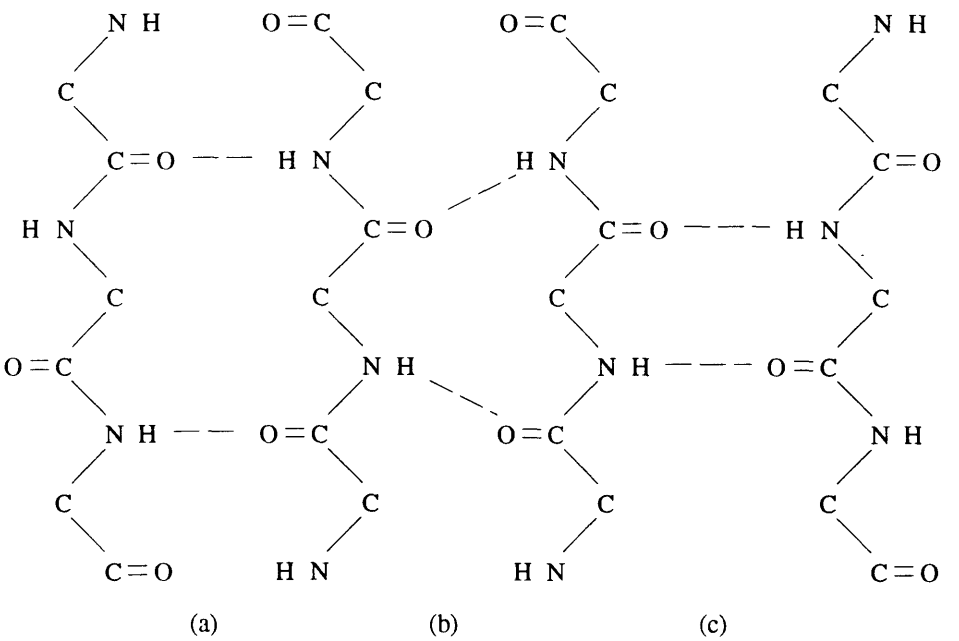


Figure 2-1: Bridges between beta strands. (a) and (c) are antiparallel bridges; (b) is a parallel bridge. A bridge exists between two nonoverlapping stretches of three residues each of there are two hydrogen bonds characteristic of beta structure. Hydrogen bonds are shown dashed; covalent bonds are shown as solid lines; the sidechains are not represented in this diagram.

type attribute.

In the threading chapter (Chapter 8), I consider a representation in which each residue has an attribute representing its local sequence. This is a specialization of the 20-valued amino acid type attribute. For a local sequence window of 13, for example, there are 20^{13} possible values of this attribute.

2.1.2 Attributes of residue pairs

I consider a number of different residue pair attributes.

Sidechain contact

I calculate whether or not the sidechains of the residues are in contact. There are two values for this attribute: **in-contact** and **not-in-contact**.

Topological relationship

For pairs that occur in beta sheets, I consider the following types:

1. **Beta pairs**. My definition of a beta pair is the two central residues in a bridge (for a definition of bridge, see Section 2.1.1). I sometimes further specialize the beta pairs into **parallel** and **antiparallel** beta pairs.

2. **Diagonal pairs**. If (i, j) and $(i+2, j+2)$ are both β_P pairs, then $(i, j+2)$ and $(i+2, j)$ are diagonal pairs (denoted by δ_P). If (i, j) and $(i+2, j-2)$ are β_A, then $(i, j-2)$ and $(i+2, j)$ are δ_A. I sometimes further specialize these into **parallel** and **antiparallel** diagonal pairs.

3. **$(i, i+2)$ pairs**. Residues i and $i+2$ are in a beta strand.

2.2 Contingency table analysis

This section provides an overview of contingency table analysis. There are a number of books on the subject; I have found the Wickens text to be particularly useful [Wickens, 1989]; Fienberg's text is also informative [Fienberg, 1977].

2.2.1 Contingency tables

Contingency table analysis is used to analyze counts of objects or occurrences, where each object has several attributes. It is well-suited for analyzing counts of residue occurrences in a set of known-structure proteins.

My data describes residues or residue pairs in protein structures. Each residue or pair has several attributes that describe its sequence and structure characteristics. Each of these attributes (such as secondary structure) has several possible values (such as alpha, beta, or coil), and each object has one and only one value for each attribute. In the statistics literature, data composed of objects with such attributes is called *categorical data.* If data objects have several attributes, the data is called *cross-classified* categorical data. Tables of counts of cross-classified categorical data are called *contingency tables.*

For example, to examine the relation between amino acid type, solvent exposure, and secondary structure type, I tabulate, for each residue in a protein, the residue's amino acid type, its exposure to solvent, and the type of secondary structure in which it appears. Thus I generate a three-dimensional contingency table. Each dimension of the table corresponds to one of the attributes.

2.2.2 Questions asked in contingency table analysis

Contingency table analysis can be used to answer questions about conditional independence and the strength of statistical relationships between variables. In this thesis, the following questions exemplify the issues that can be addressed using this statistical technique:

- Is amino acid type related to solvent exposure? In other words, are there statistically significant preferences of some amino acid types for being buried or exposed?

- Are solvent exposure and secondary structure jointly more predictive of amino acid type than either alone?

- Which is stronger: the correlation between amino acid type and solvent exposure, or the correlation between amino acid type and secondary structure?

- Are there some pairs of amino acids that show a preference to occur in neighboring positions in the protein structure? If so, is this preference dependent on the structural attributes of the residue positions in the protein?

2.2.3 Models of data

In contingency table analysis, a *model* of the data is used to create a table of expected counts. This expected count table is compared to the table of observed counts, and a standard statistical test is used to determine whether there is a significant difference between the expected and actual counts. The null hypothesis asserts that the model explains the data; the statistical test can determine if this hypothesis is false. It cannot prove that the model does fit the data, because it might be that there is not enough data to make the determination.

A model can be compared not only to the observed data, but also to another model. This is useful in determining relationships between various elements of the model.

There are many possible models for the data in a contingency table. I use a type of model called *hierarchical loglinear* models. These models are nice because they can be used to frame the types of questions that I would like to ask about the data. Hierarchical loglinear models are explained in more detail below, in Section 2.2.5.

There are many possible hierarchical loglinear models for a given contingency table. It is important, therefore, to have specific questions in mind when performing the analysis.

The meaning of the models and the relationships between the models must be supplied by the person doing the analysis. The results of contingency table analysis are often interpreted to support an assertion about causality. The statistical analysis does not intrinsically say anything about which attribute "causes" which other, but rather makes a simpler statement about the co-occurrence of events. Any meaning

	Price		
Color	low	high	total
red	N_{LR}	N_{HR}	N_R
green	N_{LG}	N_{HG}	N_G
total	N_L	N_H	N

Table 2.1: Form of a two-dimensional contingency table.

must be imposed by the person interpreting the results.

2.2.4 A simple contingency table example

Suppose we want to analyze the attributes of different apples. We have a large barrel of apples on which we will base our analysis. Each apple in the barrel has the following attributes: color and price. Each attribute has a set of possible values, as follows.

- Color: red, green.
- Price: high, low.

To build a contingency table, we tip over the barrel and record the number of apples for each possible combination of attributes. In this case there are four (low-priced green, high-priced green, low-priced red, and high-priced red). We end up with Table 2.1, which has one cell for each attribute combination.

The total number of apples with the various color and price attributes are written in the margins of the table. The cell which is the margin of the margins contains the total number of counts, N. Our model of independence generates a table of counts in which the margin totals are the same as in the observed table, but the entries in the table are computed from the marginal counts, or totals. This is because the margin counts are what we use to determine the independent probabilities of color and price.

Testing for independence

How can we test the hypothesis that an apple's color is independent of an apple's price? We estimate the probability that an apple is red from the observed frequency

of occurrence of red apples:

$$P(R) = N_R/N,$$

where N is the total number of apples and N_R is the number of red apples. The probability that an apple has a high price is estimated as

$$P(H) = N_H/N,$$

where N_H is the number of high-priced apples. If the color and price attributes are independent, then we expect that

$$P(R \text{ and } H) = P(R)P(H).$$

The expected number, E of high-priced red apples is $E(R \text{ and } H) = P(R)P(H)N$. Similarly, we can compute the expected number of low-priced red apples, high-priced green apples, and low-priced green apples.

To test for independence of color and price, we compare the expected numbers to the observed numbers. If they're very different, then our model is probably wrong, and color and price are related. If the expected and observed numbers are close to each other, then it could be that color and price are independent, and the small differences we see are due to noise in the data.

Statistics provides us with several standard tests to compare observed and expected numbers and determine whether they are "different." I use the likelihood ratio test statistic, G^2, which is a measure of the difference between the observed and expected counts. G^2 is computed for each cell in the counts table, and then summed over all cells. The formula is $G^2 = \sum(O \log(O/E))$, where O is the observed number of counts and E is the expected number of counts. It has been shown that if you look at many contingency tables, each generated by a particular model, then the distribution of G^2 values that you observe is a χ^2 distribution. Therefore, we can determine the probability that our model generated the observed data by comparing the G^2 value to a χ^2 distribution. If this probability is very small then we reject the

	Price		
Color	low	high	total
red	395	520	915
green	518	406	924
total	913	926	1839

Table 2.2: Two-dimensional contingency table for a simple example.

null hypothesis that the data was generated by our model. If the probability is large, then we don't know whether this is because our model fits the data well, or because we don't have enough data to make the determination that the model is different. Even if a model fits the data well, it could be that a simpler model would also fit well.

In order to compare the G^2 value to a χ^2 distribution, we need to know the number of degrees of freedom of our model. This number corresponds to the number of data cells in the counts table, minus the number of free parameters in the model.

As an example of testing the fit of a model, assume that the apples occur with counts as given in Table 2.2.

From the formulae given above, the expected counts are shown in Table 2.3. Note that the margin totals (N_L, N_H, N_R and N_G) in the expected table match those in the observed table. Also, the expected counts are not integers. The value of G^2 for this table is 30.6. There are three free parameters in the model (five parameters N_L, N_H, N_R, N_G and N, minus the two constraints that the sums of the margin totals must be N), and four count cells, which leaves one degree of freedom. A χ^2 test with one degree of freedom shows that the probability that the observed counts were generated by this model is extremely small, less than 0.0001. We conclude that apple price and color are not independent.

If we had had one-tenth as many apples in each cell in the contingency table, then the same analysis would have given us a G^2 of 3.06, which is not nearly as significant (P= 0.08). This observation illustrates the fact that the more counts you have, the more likely you are to find a significant result. The data that I analyze in the thesis has many total counts, and may violate some of the underlying assumptions of the

	Price		
Color	low	high	total
red	454.3	458.7	915
green	460.7	465.3	924
total	913	926	1839

Table 2.3: Expected counts for a simple example.

Taste	sweet		sour	
Price	low	high	low	high
Color				
red	202	410	193	108
green	105	197	415	209

Table 2.4: Observed values for three-way apple contingency table

statistical tests, and therefore it may be that apparent statistical significance doesn't reflect real significance. One way to deal with this problem is to ask questions about the relative size of the test statistic, as opposed to the absolute size. I will do this when I compare a series or hierarchy of models to determine which variable combinations are most influential.

Testing for conditional independence

A further exploration of our example will illustrate testing for conditional independence. Our apples have another attribute, taste. Table 2.4 shows the three-dimensional contingency table that includes the taste attribute. Table 2.2 can be obtained from Table 2.4 by summing over the two possible values of the attribute taste.

Now let's ask whether price is independent of color, but conditioned on taste. The way to do this is to separate the sweet apples from the sour apples, and within each taste group, build an table of expected counts for price vs. color. These expected tables are built based on the assumption of independence of price and color within each taste group, and are computed the same way we computed the expected counts

Taste	sweet		sour	
Price	low	high	low	high
Color				
red	205.6	406.4	197.8	103.2
green	101.4	200.6	410.2	213.8

Table 2.5: Expected values for three-way apple contingency table.

for price and color before. The expected counts are given in Table 2.5. The margin totals for taste, price, and color match those of the observed table. In addition, the two-dimensional margin totals (taste vs. price) and (taste vs. color) are the same as those in the observed table. G^2 is 0.79; the probability that the observed data was generated by this model is 0.67. Thus the model of conditional independence of price and color, relative to taste, is a very good one for the data. An interpretation could be that color influences price only indirectly through taste; all the important information in the problem can be summarized in the interaction between taste and the other variables.

This example points out important caveats about contingency table analysis. First, the design of the experiment is very important. We must try to include all the important variables (like taste) in our description of the data. Secondly, we must be careful to set up our analysis to catch the important relationships between the variables. With the three-dimensional contingency table, we can use models to support either one of our hypotheses (dependence of price and color; conditional independence of price and color). For any independence test we'd like to perform, we need to determine the appropriate context (conditioning variables) in which to do it. The best approach is to condition upon the variables which show the strongest degree of association with the variables in question [Wickens, 1989]. In our apple example, a preliminary analysis (using G^2 values) would tell us that the association between price and color is much weaker than the association between price and taste or between color and taste.

2.2.5 Loglinear models

The models I use to analyze my contingency table data are called loglinear models because they can be represented as the sum of parameters which are related to logarithms of frequencies. I use a subset of loglinear models called hierarchical loglinear models because they are easy to build and they are powerful in representing hypotheses about the data. With hierarchical loglinear models, I can pose questions about conditional independence and about the relative importance or amount of association between variables.

To cast our model of independence in loglinear style, we rewrite the expected number of high-priced red apples,

$$E(R \text{ and } H) = N\frac{N_R}{N}\frac{N_H}{N},$$

as

$$\log(\mu_{RH}) = \lambda + \lambda_{C(R)} + \lambda_{P(H)}.$$

μ_{RH} is $E(R \text{ and } H)$. $\lambda = \log(N)$ contains the information about the total number of counts in the contingency table. $\lambda_{C(R)} = \log(N_R/N)$ is the log of the frequency of red apples in the table. The "C" in the subscript stands for color, and "R" stands for red. $\lambda_{P(H)} = \log(N_H/N)$ is defined similarly.

There are five parameters in the loglinear model of color-price independence: λ, $\lambda_{C(R)}$, $\lambda_{C(G)}$, $\lambda_{C(L)}$, and $\lambda_{C(H)}$. They are further constrained by the requirement that the margin totals sum to N. This leaves three free parameters.

An example of a loglinear model for a three-dimensional contingency table is

$$\log(\mu_{ijk}) = \lambda + \lambda_{A(i)} + \lambda_{B(j)} + \lambda_{C(k)} + \lambda_{AB(ij)}.$$

μ_{ijk} is the expected count in cell (i, j, k) of the table. A indicates the table's first dimension, B the second, and C the third. If there are n_A values for the A attribute, then there are n_A λ_A parameters, one for each value. The last parameter in the model, $\lambda_{AB(ij)}$, is a function of the values i and j of attributes A and B; there

are $n_A n_B$ different parameters in the λ_{AB} set. Thus, in this model, every count is estimated by a product of terms (exponentiating the above equation). One of these terms is common to all cells in the table. Another, $e^{\lambda_{A(i)}}$ is the same for all counts that have the ith value of parameter A. Another, $e^{\lambda_{AB(ij)}}$, is the same for all counts that have the ith value of parameter A *and* the jth value of parameter B. And so on. Thus the form of the model is closely related to the row structure of the table.

There is ambiguity in the model parameters because you could get the same estimate by subtracting an amount from one parameter and adding that amount to another. Therefore, it is common to impose the constraint that the sum of the parameters in a set be 0; for example, $\sum_i \lambda_{A(i)} = 0$.

Hierarchical loglinear models require that all attributes in a high-order parameter appear in every lower-order combination. For example, if you have $\lambda_{AB(ij)}$ in your model, then you must also have $\lambda_{A(i)}$ and $\lambda_{B(j)}$. λ must appear in every model.

The λs give us information about the sign and magnitude of the effects of the attribute values in the model.

2.2.6 Margin counts

The next question is how to determine the λ parameters of a loglinear model. The optimal model is defined as the one whose parameter settings maximize the likelihood of the data given the model. It turns out that the expected counts as computed by the optimal model have the same margin totals as do the observed counts, for those margins which correspond to parameters of the model. So, for example, if $\lambda_{A(i)}$ is a model parameter, then the margin totals corresponding to attribute A will be the same in the expected count table as in the observed count table. If $\lambda_{AB(ij)}$ is a model parameter, then the two-dimensional margin computed by summing over all attributes other than A and B will be maintained.

Often, loglinear models are discussed in terms of the margins that they fix, which correspond to their parameters. Thus [AB] represents the two-dimensional margin corresponding to the parameter set λ_{AB}. The model described above for a three-dimensional table can be referred to as [AB][C]. Only the higher-order margins are

mentioned; because we know the model is hierarchical, all subsidiary margins ([A] and [B], in this case) are assumed to also be present.

A margin can also be thought of as a combination of attributes, as a term in the model, or as a parameter set.

2.2.7 Computing model parameters

For two-dimensional tables, it is straightforward to determine the expected counts, as we saw in our apple example. For three-dimensional tables, the solution is not always analytically determinable, but there exist fast iterative solutions. I used the Splus statistics program to perform these calculations [Becker *et al.*, 1988].

2.2.8 Examining conditional independence with loglinear models

The hypothesis that attributes A and B are independent corresponds to model [A][B]. The hypothesis that attributes A and B are conditionally independent, conditioned on attributes in the set $\mathcal{S}$, corresponds to the model [A$\mathcal{S}$][B$\mathcal{S}$].

2.2.9 Comparing association strengths of variables

One type of question we can ask with contingency table analysis is, "Do variables A and B interact more strongly than do variables A and C?" For example, in Chapter 3, I ask, "Do amino acid type and solvent exposure correlate more strongly than do amino acid type and secondary structure?" In fact, by building a hierarchy of loglinear models, it is possible to assign relative importance to a set of variable interactions (which correspond to margins of the contingency table).

Unfortunately, the word "hierarchy" is used in two ways in this area. The models themselves are hierarchical in the sense that higher-order margins are included only when all related lower-order margins are also included. Here, though, I am discussing a hierarchy of models, which is a series of models in which each model includes all of the margins of its predecessor, and adds more.

The relative importance of two models is judged as the relative size of the likelihood test statistic G^2. The model hierarchy approach is used by Bryant and Lawrence in their study of pairwise interactions of amino acids [Bryant and Lawrence, 1993]. They use this method to compare the relative importance or contribution to their pseudopotential of singleton and pairwise terms, concluding that singleton terms are twice as important as pairwise terms.

To report the results of such an analysis, I show the information about each model in a table or a diagram.

The models increase in complexity from top to bottom in the table. The more complex ones model the data better, but require more parameters to do so. I provide the following information about each model in the table:

- Model name.
- Terms added to previous model in hierarchy. The terms are expressed as margins. For example, [12] would be the two-dimensional table of marginal counts of the first two variables. $[A_1E_1D]$ would be the three-dimensional table of marginal counts of variables A_1 (used later for amino acid type of the first residue in a pair), E_1 (the solvent exposure of the first residue in a pair), and D (the direction, parallel or antiparallel, of the connection between beta residues).
- Significance test statistic. This statistic expresses the difference between the expected counts for this model and the observed counts, summed over the whole table. I use G^2 for examining model hierarchies, because it has nice properties of additivity for different orders of adding terms to models.
- Degrees of freedom. This number is reported as the number of unconstrained degrees of freedom in the model. Each constrained degree of freedom corresponds to a parameter in the model. The total number of degrees of freedom inherent in the contingency table is related to the number of cells (and therefore to the complexity of the protein representation). A larger number indicates more counts, and higher representational complexity. To model the observed

counts exactly, all the degrees of freedom are "used" by the model, and so the number reported here is 0. Less complicated models leave more degrees of freedom unconstrained. We expect that models that use more degrees of freedom can better fit the observed data, and take this into account when performing significance tests. The number of degrees of freedom of a table is computed keeping in mind that the tables often have symmetries. For example, there is no special meaning to which of the two members of a residue pair is the "first" residue.

- Change in significance test statistic, ΔG^2. This is measured from the previous model to the current model, and gives us an idea of how much better the current model is at modeling the data than the previous model.

- Percent change in significance test statistic. The total change in G^2 is taken from the first model reported in the table to the last model in the table. Then ΔG^2 for the current model is divided by the total change to obtain the percent change in G^2.

Because it is possible to add terms in different orders, we must be careful that adding term A first wouldn't lead us to a different conclusion than adding term B first. Sometimes the order of adding terms is restricted by the requirement that all low-order combinations of terms be added before (or at the same time) that higher-order terms containing them are added. But other times either term can be added first. In this thesis, where there is the possibility of adding either term first, I sometimes try both approaches to make sure that the conclusions I reach are not dependent upon the order of adding terms. In this case, I report two different tables, with different orders of adding terms. In general, as stated above, it is a good idea to add the most influential term first. If some attributes are fixed by design, those might be added first. On the other hand, a particular hierarchy of models might correspond in a natural way to an interpretation that we're interested in investigating. An example of this last situation can be found in Section 4.3.2, where I build a model hierarchy to correspond to the different terms in a pseudopotential function.

In Chapter 3, I examine a three-dimensional contingency table. For this table, it is easy to display all the models, from the complete independence model to the exact model of the data, in a diagram. There are six different ways to order adding the terms in the hierarchy, and so a diagram is more concise than a set of six tables, each of which can only express a single path through the hierarchy.

2.2.10 Examples of questions

In this section, I'll list some examples of questions I address using contingency analysis, along with a brief description of how I answer them.

- *Is amino acid type correlated with solvent exposure?* To answer this question, I can look at a two-way contingency table of counts categorized by amino acid type and solvent exposure. I can also look at higher-dimensional contingency tables which include this information as two of the residue attributes. (This is convenient when I want to ask other questions involving more attributes of the same data set, and possibly compare the results to those I get in answer to this question.) In particular, I look at the difference in G^2 when I add the margin [AE] (the two-dimensional marginal counts corresponding to amino acid type and solvent exposure) to a reference model. The reference model represents independence of the attributes, and so contains only the separate marginal counts [A] and [E], but not [AE]. If ΔG^2 passes a significance test then I conclude that there is correlation of amino acid type and solvent exposure.

- *Do amino acid type and solvent exposure correlate more strongly than do amino acid type and secondary structure?* This question is addressed in Chapter 3. To answer it, I analyze the three-way contingency table of amino acid type, solvent exposure, and secondary structure at single residue positions in the protein structures. I want to compare the ΔG^2 values for adding the pairwise margins [AE] (amino acid type and solvent exposure) and [AS] (amino acid type and secondary structure) to a reference model in which all variables are considered to be independent ([A][S][E] are the margins in the independent model).

- *Does the correlation between amino acid type and solvent exposure contain information that is also contained in the correlation between amino acid type and secondary structure?* It might be possible, for example, that the preferences shown by amino acids for particular types of secondary structure could be completely explained by their solvent exposure preferences. This would be useful to know in designing a knowledge representation because we wouldn't have to represent amino acid preferences for secondary structure at all. To address this question, I consider adding the margins [AE] and [AS] serially, in both possible orders, to the reference model. I look at the resulting ΔG^2 values for both model series.

- *How much of the apparent pairwise association between amino acids is due to the exposure preference of each amino acid?* This question is addressed in Section 6.3.6. It is an interesting question because it suggests that a singleton term (association between amino acid and environment) might explain a large portion of a pairwise term (association between two amino acids). To answer this question, I look at two margins, or terms. One is $[A_1A_2]$, which corresponds to the association between amino acid types. The other is $[E_1E_2]$, which corresponds to the association between exposure types. From a reference model, I add margin $[A_1A_2]$ and get a change in likelihood ratio test statistic ΔG^2_{AA}. From the same reference model, I add margin $[E_1E_2]$ and get ΔG^2_{EE}. Comparing ΔG^2_{AA} and ΔG^2_{EE} gives me some idea of the relative importance, although it is also important to consider the number of degrees of freedom corresponding to each. I also ask what happens when I add first $[A_1A_2]$ and then $[E_1E_2]$ in a chain of three models, starting from the reference model. I compare that to adding the margins in the reverse order.

Chapter 3

Single-Residue Statistics

3.1 Introduction

In this chapter I use contingency table analysis to examine the relationship between amino acid type and the structural parameters associated with the position of a residue in protein's structure. These parameters are the residue's exposure to solvent and secondary structure type. I am particularly interested in the relative importance of solvent exposure and secondary structure. I show that solvent exposure is more strongly correlated with amino acid class and with secondary structure than is amino acid class with secondary structure. This is true regardless of whether amino acids are grouped into three classes by hydrophobicity class, or left as 20 separate types.

3.2 Method

3.2.1 Data

I analyzed the 252 proteins listed in Section B.2.2. This is a subset of the set of proteins compiled by Hobohm and colleagues in their effort to identify protein sequences that have low sequence homology [Hobohm *et al.*, 1992]. I chose all proteins in the list for which Kabsch and Sander's DSSP secondary structure files exist. To determine the residue attributes, I used the publicly available DSSP files of Kabsch

Class	Residues
Hydrophobic	VLIFYWMC
Neutral	TSAGP
Polar	KRDNHEQ

Table 3.1: Amino acid classification into three hydrophobicity classes.

and Sander [Kabsch and Sander, 1983].

3.2.2 Residue attributes

Each residue has three attributes: amino acid type, solvent exposure, and secondary structure.

I considered two different versions of the attribute amino acid type, A. I built two contingency tables, one for each attribute definition. In the first definition, each of the 20 amino acid types was its own class. In the second definition, the amino acids were grouped into three classes by their hydrophobicity as shown in Table 3.1.

Relative solvent exposure, E, was computed for each residue by dividing DSSP's "ACC" accessibility number by the maximum accessibility for that residue type (see Section B.3). Residues with less than 20% relative solvent exposure were considered buried; the others exposed.

I used the DSSP secondary structure assignments to determine the secondary structure attribute, S, of each residue (alpha = "H"; beta = "E"; coil = anything else) [Kabsch and Sander, 1983].

3.2.3 Contingency table analysis

A contingency table analysis using loglinear models was performed to determine the interactions between amino acid type or hydrophobicity class, solvent accessibility, and secondary structure. This approach looks at the difference between observed and expected counts (as measured by the likelihood test or Pearson statistic G^2, which has a χ^2 distribution) across models to determine how much of the error is accounted

for by which combinations of variables.

Each pair of models analyzed differs by the inclusion or exclusion of one marginal term. Each marginal term corresponds to a set of variables; the absence of a term in a model indicates an assumption of independence among those variables. Thus, loglinear analysis tests for the presence and strength of conditional dependence among a model's variables, as well as the power of each of the variables and combinations thereof to explain the observed data.

Two three-way contingency tables were built, one with all 20 amino acids classified separately, and another with the amino acids grouped into three classes. Thus each residue was assigned to cell (i, j, k) in the contingency table, where $i \in$ A= {hydrophobic, neutral, polar} or {A,C,D,E,F,G,H,I,K,L,M,N,P,Q,R,S,T,V,W,Y}; $j \in$ E = {buried, exposed}; and $k \in$ S = {alpha, beta, coil}. As described in Section 2.2.9, I then built all possible loglinear models containing all one-factor effects, and determined the likelihood ratio test statistic for each nested pair.

3.3 Results and Discussion

Table 3.2 shows the contingency tables. There are 53,037 total residues. Marginal totals for each amino acid or amino acid group appear down the side. The three-way table is presented in two dimensions, and so the two-dimensional margins totals for solvent exposure and secondary structure are shown at the bottom of the tables. Marginal totals for secondary structure and solvent exposure are given in the caption. The residues are approximately evenly distributed between buried and exposed. 30% of the residues are alpha; 21% are beta; 49% are coil.

In the grouped amino acid table, counts range from 930 (exposed beta hydrophobic) to 7008 (exposed coil polar). In the full $20 \times 3 \times 2$ table, the counts range from 40 (Met beta exposed) to 1864 (Gly coil exposed).

3.3.1 Loglinear models

	Alpha		Beta		Coil		
	Buried	Exposed	Buried	Exposed	Buried	Exposed	Total
H	4,640	1,156	4,727	930	3,967	2,381	17,801
N	2,514	1,726	2,098	950	4,062	6,582	15,790
P	1,569	4,044	1,079	1,580	2,024	7,008	17,304
Total	8,723	6,926	7,904	3,460	10,053	15,971	53,037

G	404	171	528	110	1,213	1,864	4,290
P	145	163	123	100	740	1,223	2,494
D	224	575	172	170	462	1,536	3,139
E	283	1,027	156	329	228	1,155	3,178
A	1,243	644	587	110	896	1,036	4,516
N	211	327	163	143	457	1,165	2,466
Q	236	481	140	181	196	664	1,898
S	341	408	389	241	607	1,390	3,376
T	381	340	471	389	606	1,069	3,256
K	160	895	119	406	154	1,370	3,104
R	301	587	164	279	271	773	2,375
H	154	152	165	72	256	345	1,144
V	801	199	1,206	231	728	469	3,634
I	747	166	888	130	552	323	2,806
M	365	99	240	40	230	168	1,142
C	152	41	238	48	304	191	974
L	1,420	311	944	167	957	516	4,315
F	563	114	565	108	543	259	2,152
Y	385	172	468	162	432	361	1,980
W	207	54	178	44	221	94	798
Total	8,723	6,926	7,904	3,460	10,053	15,971	53,037

Table 3.2: Three-way contingency table of observed counts for amino acid group, solvent exposure, and secondary structure. H: hydrophobic; N: Neutral; P:Polar. Total counts for the secondary structure attribute are alpha 15,649; beta 11,364; coil 26,024. Total counts for the solvent exposure attribute are buried 26,680; exposed 26,357.

Model	Grouped G^2	df	Ungrouped G^2	df	marginal terms						
[A]	19,745	15	23,998	100	A						
[E]	19,756	16	33,664	118		E					
[S]	13,495	15	27,404	117			S				
[A][E]	19,743	14	23,996	99	A	E					
[A][S]	13,483	13	17,736	98	A		S				
[E][S]	13,493	14	27,402	116		E	S				
[A][E][S]	13,481	12	17,734	97	A	E	S				
[A][ES]	10,133	10	14,386	95	A	E	S			ES	
[AE][S]	5,040	10	7,974	78	A	E	S		AS		
[E][AS]	10,756	8	12,228	59	A	E	S	AE			
[AE][AS]	2,315	6	2,468	40	A	E	S	AE	AS		
[AE][ES]	1,692	8	4,626	76	A	E	S	AE		ES	
[AS][ES]	7,408	6	8,880	57	A	E	S		AS	ES	
[AE][AS][ES]	157	4	306	38	A	E	S	AE	AS	ES	
[AES]	0	0	0	0	A	E	S	AE	AS	ES	AES

Table 3.3: Loglinear models built from the contingency tables whose dimensions are amino acid (grouped by hydrophobicity or ungrouped), solvent accessibility, and secondary structure. The variables are (1) amino acid (A), (2) solvent exposure (E), and (3) secondary structure (S). G^2 is the likelihood ratio test statistic, and df is the number of degrees of freedom. The marginal terms in each model are indicated.

Table 3.3 describes the models built from the contingency tables. For each model, the likelihood ratio test statistic G^2, the number of degrees of freedom, and the marginal terms of the model are shown.

Model [A][E][S] fits all three one-dimensional margins. In other words, the total numbers of each category for A, E, and S are the same in the observed and predicted tables; for example, $\sum_{j,k} E_{[A][E][S]}(i,j,k) = \sum_{j,k} N(i,j,k) = M_i$, where $E(i,j,k)$ is the expected value in cell (i,j,k), and $N(i,j,k)$ is the corresponding observed count. This model corresponds to the assumption of independence between the variables. The expected values for [A][E][S] are:

$$E_{[A][E][S]}(i,j,k) = \frac{M_i}{N}\frac{M_j}{N}\frac{M_k}{N}N = \frac{M_i M_j M_k}{N^2},$$

where M_i is the marginal total for the i'th category of the first variable, A, and so on.

Model [A][ES] maintains the one-dimensional margins [A], [E], and [S], and also the two-dimensional margin [ES]. Including the two-dimensional margin means that

$$\sum_i E_{[A][ES]}(i,j,k) = \sum_i N(i,j,k).$$

The difference between models [A][E][S] and [A][ES] is that the latter does not assume that margins [E] and [S] are independent. Thus by comparing these models we test the independence of variables E and S. The difference likelihood ratio test statistics, ΔG^2, for two models, one a generalization of the other, is distributed as χ^2, with Δdf degrees of freedom, and thus we can test the null hypothesis that the two models are equivalent.

3.3.2 Model hierarchies

Figures 3-1 and 3-2 show the model hierarchies. The path marked with heavy arrows shows the model sequence which explains the most variance the most quickly. The order is different in the two cases; for the grouped amino acids, the order is (1) [AE]

Model	G^2	df	added	ΔG^2	$\%\Delta G^2$	Δdf
[A][E][S]	13,481	12				
[AE][S]	5,040	10	[AE]	8,441	62.6	2
[AE][ES]	1,692	8	[ES]	3,348	24.8	2
[AE][ES][AS]	157	4	[AS]	1,535	11.4	4
[AES]	0	0	[AES]	157	1.2	4
Total				13,481	100.0	12

Table 3.4: Singleton model hierarchy for grouped amino acids.

(ΔG^2 = 8,441, Δdf = 2), (2) [E]] (ΔG^2 = 3,348, Δdf = 2), (3) [AS] (ΔG^2 = 1,535, Δdf = 4). On the other hand, when the 20 amino acids are considered separately, the order is (1) [AE] (ΔG^2 = 9,760, Δdf = 19), (2) [AS] (ΔG^2 = 5,506, Δdf = 38), (3) [ES] (ΔG^2 = 2,162, Δdf = 2). In both cases, the most variance is explained by the [AE] term, the association between amino acid and solvent accessibility. In each hierarchy, each model explains significantly more observed data than the one before it; in other words, all variable interactions are significant. Even the no-three-factor model, [AE][AS][ES], is significantly different from the full [AES] model; thus, there exist nonnegligeable three-way effects between A, E, and S.

Model hierarchy for 3-class amino acid representation

Table 3.4 shows more details about the model sequence which explains G^2 the most quickly, for grouped data. Only 11.4% of the total change in G^2 is due to the pairwise association between amino acid type (hydrophobicity class) and secondary structure.

It is instructive to examine the model parameters for these models. These are shown in Table 3.5. The constraint that parameters in a set sum to 0 is easy to see. The exact values of the lower-order parameters depend slightly on which higher-order parameters are included in the model. The λ parameter can be thought of as representing (approximately) the log of the average cell count. To determine the expected count, this average cell count is modified by the higher-order parameters.

The one-dimensional λ parameters correspond in a natural way to the overall ratios of each attribute value. For example, the data set is 30% alpha, 21% beta, and

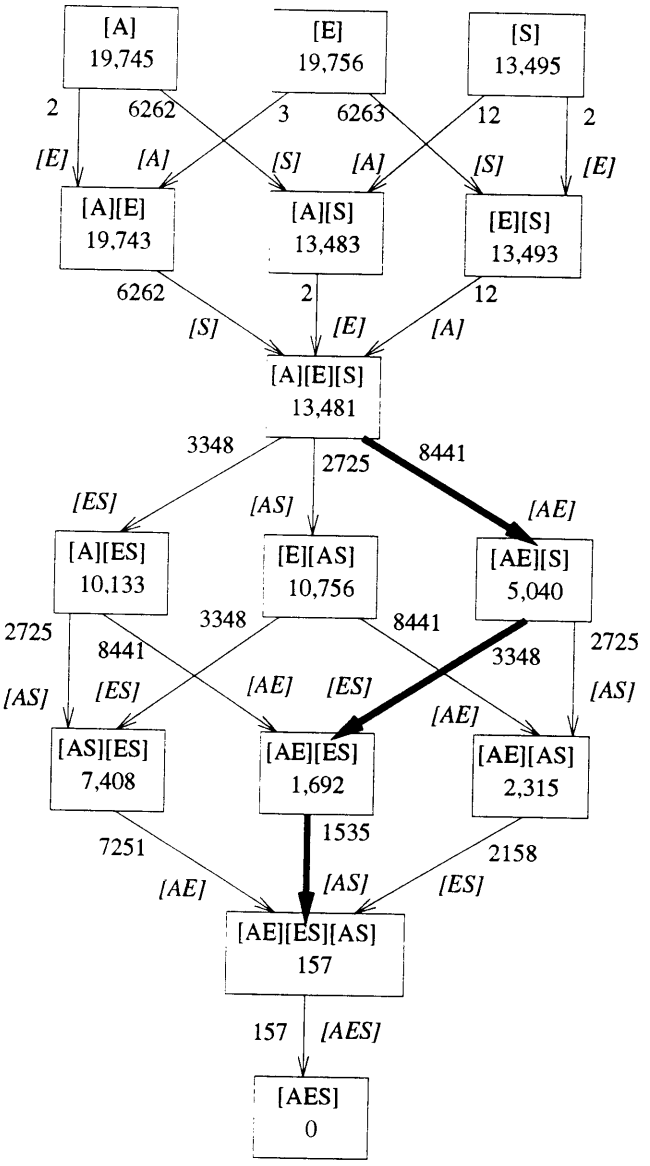


Figure 3-1: Nested model hierarchy for the three-way contingency table whose categories are amino acid class (variable 1), solvent exposure (variable 2), and secondary structure (variable 3). Each box represents a model. The model name and likelihood ratio statistic, G^2, are listed in the box. Arrows are drawn between models related by the addition of marginal terms. The arrows are annotated with the difference in G^2 and (in italics) the added marginal terms. Heavy arrows indicate the nested hierarchy which explains the most variance the earliest.

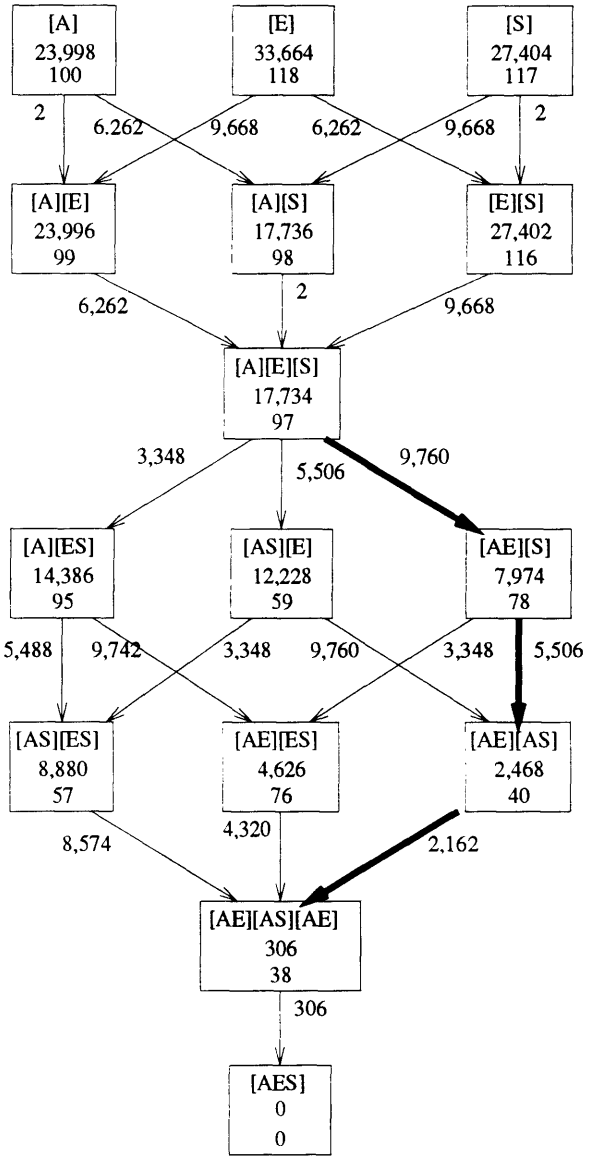


Figure 3-2: As previous figure, except for 20 amino acid types instead of three classes. Degrees of freedom of each model are shown at the bottom of the box.

Model	λ	λ_A	λ_E	λ_S	λ_{AE}		λ_{AS}			λ_{ES}		
[A][E][S]	7.93	0.01	0.01	-0.06								
		0.01	-0.01	-0.38								
		-0.02		0.45								
[AE][S]	7.48	-0.05	0.01	-0.06	0.54	-0.54						
		0.10	-0.01	-0.38	-0.04	0.04						
		-0.05		0.45	-0.50	0.50						
[AE][ES]	7.80	-0.05	0.10	-0.03	0.54	-0.54				0.02	0.31	-0.33
		0.10	-0.10	-0.43	-0.04	0.04				-0.02	-0.31	0.33
		-0.05		0.46	-0.50	-.50						
[AE][AS][ES]	7.79	0.02	0.09	-0.03	0.50	-0.50	-0.00	0.25	-0.25	0.04	0.25	-0.30
		0.03	-0.09	-0.43	0.00	-0.00	-0.15	-0.12	0.27	-0.04	-0.25	0.30
		-0.05		0.46	-0.50	0.50	0.15	-0.13	-0.02			

Table 3.5: Model parameters for singleton model sequence, grouped amino acids. The three components to the λ_A vector are H, N, and P. Those of the λ_E vector are buried and exposed. Those of the λ_S vector are alpha, beta, and coil. The two-dimensional parameters are displayed with the first subscript indicating the attribute that indexes the rows and the second subscript indicating the attribute that indexes the columns.

49% coil. The average percentage would be 33.3%. There are slightly fewer alpha residues than expected at random, so the $\lambda_{S(A)}$ parameter, -0.06, is slightly negative. There are far more coil residues than expected at random, so the $\lambda_{S(C)}$ parameter, 0.45, is positive. In fact, $33\% \times e^{0.45} = 52\%$, which is approximately the composition of coil in the data set. Because the residues are approximately evenly split between buried and exposed, the λ_E vector values are close to 0.0.

From the two-dimensional parameter values, we can see the interaction of attributes. For example, hydrophobic residues prefer to be buried ($\lambda_{AE(H,\text{buried})} = 0.54$). Buried coil residues are disfavored ($\lambda_{ES(\text{buried,coil})} = -0.30$).

Similar information about the relationship of specific attribute values can be determined by looking at the ratio of the expected values of two models in the hierarchy. This is done in Table 3.6. Tables are collapsed only when all the values for the individual cells are equal.

From the [AE][S]/[A][E][S] table, we can see the effect of adding the [AE] term. Buried hydrophobes and exposed polars are favorable and can now be modeled. In the [AE][ES]/[AE][S] table, in which the [ES] margin is added, it is apparent that alpha and beta prefer to be buried, while coil prefers to be exposed. In the

	E	
A	buried	exposed
H	1.49	0.50
N	0.96	1.00
P	0.54	1.50

(a)

	E	
S	buried	exposed
alpha	1.10	0.89
beta	1.40	0.61
coil	0.77	1.20

(b)

E	buried			exposed		
S	alpha	beta	coil	alpha	beta	coil
A						
H	1.03	1.21	0.81	1.09	1.53	0.85
N	0.85	0.79	1.30	0.75	0.84	1.14
P	1.18	0.80	0.99	1.15	0.93	0.95

(c)

E	buried			exposed		
S	alpha	beta	coil	alpha	beta	coil
A						
H	1.03	0.99	0.98	0.90	1.04	1.04
N	1.05	1.03	0.96	0.94	0.93	1.03
P	0.87	0.97	1.16	1.06	1.02	0.96

(d)

Table 3.6: Ratios of expected values in model hierarchy. (a) [AE][S]/[A][E][S]. (b) [AE][ES]/[AE][S]. (c) [AE][AS][ES]/[AE][ES]. (d) [AES]/[AE][AS][ES].

Model	G^2	df	added	ΔG^2	$\%\Delta G^2$	Δdf
[A][E][S]	17,734	97				
[AE][S]	7,974	78	[AE]	9,760	55.0	19
[AE][AS]	2,468	40	[AS]	5,506	31.0	38
[AE][ES][AS]	306	38	[ES]	2,162	12.2	2
[AES]	0	0	[AES]	306	1.7	38
Total				17,734	100.0	97

Table 3.7: Singleton model hierarchy, all 20 amino acid types.

[AE][AS][AE]/[AE][ES] table, we have added the [AS] margin. Adding the interaction between hydrophobicity class and secondary structure class allows the representation of favoring polar residues in alpha structure, hydrophobic residues in beta structure, and neutral residues in coil structure. These preferences are those over and above those which can be represented indirectly by combining the effects of the [AE] and [ES] model terms. Finally, the three-way term [AES], when compared to the full two-way term model, allows the expression of a preference for buried polars in coil structure, and the disfavoring of buried polars in alpha structure as well as exposed hydrophobes and neutrals in alpha structure, and neutrals in exposed beta structure.

Hierarchy for 20-class amino acid representation

Table 3.7 shows the best model sequence for the amino acid representation in which all 20 amino acids are represented separately. The [AS] margin explains a much larger percentage of G^2. Note, however, that the [AS] margin has 38 degrees of freedom, while the [ES] margin has 2; the larger number of parameters should allow a better fit. Specializing the exposure representation might increase the relative amount of G^2 due to the [ES] margin.

3.3.3 Nonspecific vs. specific amino acid representation

Lifson and Sander compared the nonspecific (3-class) with the specific (20-class) amino acid representations in looking at pairs of residues in beta strands [Lifson and Sander, 1980]. I do something similar in this section. I compute expected counts

for the specific representation based on the nonspecific representation. I then compare these nonspecific expected counts to the observed counts. Alternatively, I examine one particular margin by computing expected counts from the ratio of two nonspecific models whose difference is that margin. I then compare these expected counts to the expected counts of a specific model with the same margin.

To compute the expected specific counts based on the nonspecific counts,

$$E'_{[AES]} = \frac{E_{[HES]}}{E_{[H][E][S]}} E_{[A][E][S]}.$$

$E'_{[AES]}$ is the expected count for a cell in the specific count table. $E_{[HES]}$ is the observed count for a cell in the nonspecific count table, where the specific amino acid A belongs to hydrophobicity class H. $E_{[H][E][S]}$ is the expected count for a cell in the nonspecific table, based on a model of independence. $E_{[A][E][S]}$ is a similar expected count, but for the specific amino acid representation.

In general, to examine a particular margin, I use

$$E'_{[Ac]} = \frac{E_{[Hc]}}{E_{[Hs]}} E_{[As]},$$

where A indicates the specific amino acid representation, H indicates the nonspecific amino acid representation, c indicates the more complex model (the one with the margin of interest), and s indicates the simpler or reference model (the one without the margin of interest).

I use the model hierarchy of Table 3.7, adding one margin at a time and looking at ΔG^2 and the ratio of specific counts to those predicted by the nonspecific model.

Amino acid and solvent exposure

I start with the association of amino acid and solvent exposure, which I have shown is the strongest pairwise association among the three attributes I have examined. I compute the expected counts using the formula described above, where the reference models are [H][E][S] and [A][E][S], and the complex models are [HE][S] and [AE][S]. I find a χ^2 value of 680.4 for 95 degrees of freedom, which is statistically significant.

Class	AA	buried	exposed
H	I	1.04	0.88
	F	1.04	0.89
	L	1.03	0.92
	W	1.01	0.96
	V	1.00	0.99
	M	0.98	1.07
	C	0.95	1.15
	Y	0.87	1.40
N	A	1.25	0.77
	G	1.03	0.97
	T	0.93	1.07
	P	0.84	1.15
	S	0.82	1.17
P	R	1.15	0.95
	N	1.25	0.91
	Q	1.12	0.96
	D	1.01	1.00
	E	0.78	1.08
	H	0.67	1.98
	K	0.52	1.18

Table 3.8: Ratios of specific to nonspecific expected counts for margin [AE]. The amino acids (AA) are grouped by class (H: hydrophobic; N: neutral; P: polar). Within each class, the amino acids are ordered by their ratios.

The ratios of predicted counts, $E'_{[AE][S]}/E_{[AE][S]}$, are shown in Table 3.8. Each ratio describes the difference between the count that would be predicted based on the nonspecific model [HE][S], and the count actually predicted by the specific model [AE][S]. A number larger than 1.0 indicates that the nonspecific model underpredicts that cell. Thus, for example, Tyr (Y) is more exposed than the average hydrophobic residue. Ala (A) is more buried than the average neutral residue. The table is ordered so that the residues which tend more toward being buried within their hydrophobicity class appear at the top of their section.

Amino acid and secondary structure

I next add the margin [AS], to look at how the association between amino acid and secondary structure is different for the specific amino acid representation than for the nonspecific amino acid representation. I leave the margin [AE], as it is the strongest effect and I want to look at [AS] without confusion from the [AE] effects. The expected counts are computed as

$$E'_{[AE][AS]} = \frac{E_{[HE][HS]}}{E_{[HE][S]}} E_{[AE][S]}.$$

The resulting model has a χ^2 of 1446.1, for 74 degrees of freedom, which is statistically significant. Table 3.9 shows the ratios of expected counts.

Within each hydrophobicity class, some residues favor alpha or beta or coil more than the average residue in that class. This is indicated in the table.

Solvent exposure and secondary structure

Adding the margin [ES] results in no significant change between the specific and nonspecific models. This is what we would expect, because the margin does not include the amino acid representation.

Three-way effect

Adding the margin [AES] gives us a model with a χ^2 of 84.4, with 34 degrees of freedom, which is significant. The results are shown in Table 3.10. These are effects that occur after all second-order interactions have been accounted for. Some of the cells which are interesting include:

- Prefer exposed: Gly (G) coil, Trp (W) beta, His (H) alpha, Pro (P) alpha, Pro (P) beta, and Arg (R) beta.
- Prefer buried: Gly (G) alpha, Gly (G) beta, Trp (W) coil, Ala (A) beta, and Met (M) beta.

Class	AA	alpha	beta	coil	Interpretation
H	M	1.25	0.77	0.98	alpha-favoring hydrophobe
H	L	1.23	0.81	0.96	alpha-favoring hydrophobe
H	V	0.85	1.24	0.92	beta-favoring hydrophobe
H	I	1.00	1.14	0.87	beta-favoring hydrophobe
H	C	0.61	0.92	1.43	coil-favoring hydrophobe
H	Y	0.86	1.00	1.12	coil-favoring hydrophobe
H	W	1.00	0.88	1.11	coil-favoring hydrophobe
H	F	0.97	0.98	1.05	coil-favoring hydrophobe
N	A	1.77	0.91	0.72	alpha-favoring neutral
N	T	0.94	1.55	0.87	beta-favoring neutral
N	S	0.94	1.10	1.00	beta-favoring neutral
N	P	0.52	0.53	1.33	coil-favoring neutral
N	G	0.57	0.87	1.21	coil-favoring neutral
P	E	1.27	0.99	0.83	alpha-favoring polar
P	Q	1.16	1.10	0.87	alpha-favoring polar
P	R	1.15	1.21	0.84	beta-favoring polar
P	K	1.05	1.10	0.94	beta-favoring polar
P	H	0.82	0.65	1.47	coil-favoring polar
P	N	0.67	0.81	1.26	coil-favoring polar
P	D	0.78	0.71	1.22	coil-favoring polar

Table 3.9: Ratios of specific to nonspecific expected counts for margin [AS]. The amino acids (AA) are grouped by class (H: hydrophobic; N: neutral; P: polar). Within each class, the amino acids are ordered by which type of secondary structure they prefer, relative to the rest of the hydrophobicity class.

AA	alpha		beta		coil	
	buried	exposed	buried	exposed	buried	exposed
V	1.01	0.97	1.00	0.98	0.99	1.02
I	1.00	1.00	1.02	0.88	0.97	1.06
M	1.00	1.01	1.03	0.84	0.96	1.07
C	1.00	1.01	1.00	0.99	1.01	0.98
L	1.00	0.99	0.99	1.04	1.00	1.01
F	1.00	0.96	0.98	1.16	1.03	0.96
Y	0.98	1.09	0.97	1.10	1.07	0.92
W	0.97	1.12	0.94	1.35	1.09	0.84
G	1.11	0.80	1.13	0.63	0.94	1.05
P	0.85	1.19	0.84	1.32	1.10	0.94
A	0.95	1.07	1.08	0.69	0.98	1.03
S	0.92	1.10	1.02	0.98	1.03	0.98
T	1.00	1.01	0.86	1.26	1.12	0.94
D	0.93	1.04	1.14	0.89	0.94	1.01
E	1.03	0.97	1.00	0.99	1.04	1.01
N	1.06	0.99	1.03	0.99	0.90	1.02
Q	1.10	0.96	1.00	1.01	0.94	1.01
K	1.07	0.95	1.00	0.99	0.97	1.03
R	1.12	0.96	0.83	1.14	1.05	0.98
H	0.86	1.29	1.04	0.90	1.09	0.92

Table 3.10: Ratios of specific to nonspecific expected counts for margin [AES].

3.4 Conclusions

3.4.1 Solvent exposure is of primary importance

The association of amino acid with solvent exposure by hydrophobicity is a stronger effect than the association of amino acid with secondary structure. This has a number of implications for structure prediction algorithms.

One of the hopes for secondary structure prediction has been that it might serve as a stepping-stone to tertiary structure determination. These results suggest that, given a choice, it might make sense to use solvent exposure rather than alpha/beta/coil as an intermediate representation in tertiary structure predictions.

The results also confirm that it is a good idea to use solvent accessibility preference in prediction methods. Currently most inverse folding or threading tertiary structure prediction methods do use solvent accessibility in their pseudopotential functions. However there are some, notably those that focus on pairwise potentials, that do not explicitly take solvent accessibility into account.

3.4.2 Grouping residues by hydrophobicity class

There are a number of reasons to generalize the representation of amino acid type. Here I reduced the attribute size from 20 values to three values, grouping the amino acids by their hydrophobicity. This technique might be useful as a means toward reducing the complexity of a knowledge representation for proteins, particularly if there were a number of other attributes. In addition, it is useful to see how the grouping affects the statistical results.

Not surprisingly, hydrophobicity class is much more strongly correlated with solvent exposure than with secondary structure.

I looked at the association between attributes that were due to the specific representation of amino acids (all 20 amino acids in their own class), over and above that due to the nonspecific representation (three groups). The association between amino acid type and secondary structure in particular is improved by the specific

representation.

3.4.3 Attribute preferences

By looking at the ratios of the expected values from related models or at the model parameters, it is possible to determine the preferences and relationships between the different attributes. Thus I have numerical values for the preferences of the various amino acid types for being buried or exposed, or in a given type of secondary structure. This information can be determined in the context of other attribute relationships. The preferences can be used in a variety of ways in prediction algorithms.

Chapter 4

Paired-Residue Statistics

4.1 Introduction

In Chapter 3, I analyzed the attributes amino acid type, secondary structure, and solvent exposure of residues in proteins. In this chapter, I use contingency table analysis to ask questions about pairs of residues. The questions include the following:

- What effects do solvent exposure and secondary structure have on amino acid pairing?
- Is there specific residue pairing over and above pairing of amino acids by hydrophobicity class?
- What is the relative importance of paired-residue and single-residue attribute associations? Single-residue associations are those between attributes of a single residue. Paired-residue associations are those between attributes of two residues. This question is relevant to the design of pseudopotential functions for threading methods of predicting protein structure.
- How does sample size affect the results?
- How can we improve the efficacy of pairwise terms in threading pseudopotentials?

I analyze counts for all pairs of residues in each of a set of proteins, taking into consideration the secondary structure, solvent exposure and hydrophobicity of each residue, as well as whether the side chains are in contact.

4.2 Methods

In this section, I describe the data used and the residue attributes for the contingency table.

4.2.1 Data

I used two different data sets. The first is listed in Appendix section B.2.4; this is a set of 49 nonhomologous and monomeric proteins.

The second data set has 248 proteins. These are the 252 listed in Appendix section B.2.2 (and used in Chapter 3), with a few changes due to the unavailability of some of the PDB files. The following four proteins on the list were not used: 1grd, 1lig, 1mrm, and 1pde. In addition the following eight substitutions were made (due to the unavailability of old PDB files): 2aai for 1aai, 2bop for 1bop, 2cas for 1cas, 3cox for 1cox, 2cpl for 1cpl, 2sas for 1sas, 3sdh for 1sdh, and 2tmd for 2tmd.

I used the smaller set first, and then the larger set, to look at the effect of sample size.

Structure files from the Protein Data Bank [Bernstein *et al.*, 1977, Abola *et al.*, 1987] were used to compute sidechain contact. Secondary structure and solvent exposure information was obtained from the DSSP files of Kabsch and Sander [Kabsch and Sander, 1983].

4.2.2 Representational attributes

I used three singleton attributes (amino acid type, solvent exposure, and secondary structure type) and one pairwise attribute (residue contact). The contingency table contains counts of residue pairs. Each pair has seven attributes: two sets of singleton

Class	Residues
Hydrophobic	VLIFYWMC
Neutral	TSAGP
Polar	KRDNHEQ

Table 4.1: Amino acid classification into three hydrophobicity classes.

attributes plus one pairwise attribute. For analysis, I split the contingency table into two separate six-dimensional marginal count tables, one for residue pairs in contact and the other for residue pairs which are not in contact.

The non-contacting side chains function as a control group; I expected analysis of these to show random association between pairs.

Amino acid type

As in chapter 3, I built separate contingency tables for each of two representations of the amino acid type. The attributes are A_1 for the first residue in the pair and A_2 for the second residue. In the first representation, each of the 20 amino acid types was its own class. In the second representation, the amino acids were grouped into three classes by their hydrophobicity as shown in Table 4.1.

Solvent exposure

I computed relative exposure for each residue, E_1 and E_2, by dividing the "ACC" accessibility number (as defined by Kabsch and Sander) by the maximum accessibility for that residue type (see Section B.3). Residues with less than 20% relative exposure were considered buried; the others exposed.

Secondary structure

I used Kabsch and Sander's secondary structure assignments to determine the secondary structure attribute of each residue (alpha = "H"; beta = "E"; coil = anything else) [Kabsch and Sander, 1983]. I call these attributes S_1 for the first residue and S_2 for the second residue.

Side-chain contact

The closest distance between two residues was computed by examining all their non-hydrogen atoms. Backbone atoms (except the alpha carbon) were also excluded from this computation). Atoms were deemed to be in contact if the closest distance from one to the other was less than the sum of their expanded radii plus 0.1 Angstrom. Radii for the atoms are listed in section B.4.

4.3 Results and Discussion

I first present an analysis of the non-contacting pairs in the set of proteins. Then I present the results for the contacting pairs, and perform analyses to try to answer the questions I set forth in the introduction.

There were 1,833,584 non-contacting pairs and 15,582 contacting pairs in the smaller data set of 49 proteins.

There were 250,174 contacting pairs, and 15,979,682 noncontacting pairs in the larger data set of 248 proteins.

4.3.1 Non-contacting pairs

This analysis was done as a control to test for the presence of apparent association between amino acid residues which are not in contact in the protein. I expected to see no correlation between attributes of pairs that were not in contact. Therefore, the degree of apparent correlation should give me an idea of the amount of "noise" in the tables. This is a more reliable baseline than relying solely on the statistical significance tests. Each amino acid in the pair is classified according to its amino acid type, secondary structure, and solvent accessibility. I first discuss the two-dimensional margin tables, and then perform contingency table analysis on the full six-dimensional table.

	Exposure		Secondary Structure		
	buried	exposed	alpha	beta	coil
hydrophobic	4,085,263	1,263,441	1,814,539	1,609,681	1,924,484
neutral	2,758,821	2,632,682	1,323,221	867,314	3,200,968
polar	1,536,064	3,703,411	1,746,274	763,156	2,730,045
alpha	2,821,091	2,062,943			
beta	2,298,876	941,275			
coil	3,260,181	4,595,316			

Table 4.2: Non-contacting pair marginal counts: singleton terms. All three two-dimensional tables show significant nonrandom association ([AE]: $G^2 = 2,455,692$ with 2 degrees of freedom, [AS]: $G^2 = 790,047$ with 4 degrees of freedom, [ES]: $G^2 = 894,805$ with 2 degrees of freedom).

	Exposure		Secondary Structure		
	buried	exposed	alpha	beta	coil
hydrophobic	1.46	0.50	1.10	1.48	0.73
neutral	0.98	1.00	0.80	0.79	1.21
polar	0.56	1.50	1.10	0.72	1.06
alpha	1.10	0.89			
beta	1.35	0.61			
coil	0.79	1.23			

Table 4.3: Non-contacting pair observed to expected: singleton terms. Expected counts were generated by the model of independence between attributes.

Two-dimensional tables of margins

The six-dimensional contingency table can be bewildering. I will start by looking at more comprehensible pieces of it. In this section I'll describe the nine two-dimensional tables of marginal totals. Three of the tables are about singleton attributes; three describe paired residues' same attribute; and three describe paired residues' different attributes.

Table 4.2 shows the three two-dimensional margin count tables that correspond to the singleton terms (association of a residue's amino acid type with its own secondary structure and solvent exposure). As we know from the previous chapter, these attributes are related. And in fact, all of these tables show significant non-random association (see the caption to Table 4.2).

Table 4.3 shows the ratios of observed to predicted values for the three singleton margins. These should correspond roughly to the singleton ratios computed in

$[A_1A_2]$	H	N	P	H	N	P
H	1,777,104	1,805,531	1,766,069	0.99	1.00	1.01
N	1,805,531	1,842,512	1,743,460	1.00	1.01	0.99
P	1,766,069	1,743,460	1,729,946	1.01	0.99	1.01

$[E_1E_2]$	buried	exposed	buried	exposed
buried	4,465,212	3,914,936	1.02	0.98
exposed	3,914,936	3,684,598	0.98	1.02

$[S_1S_2]$	alpha	beta	coil	alpha	beta	coil
alpha	1,888,282	747,618	2,248,134	1.26	0.75	0.94
beta	747,618	855,348	1,637,185	0.75	1.30	1.03
coil	2,248,134	1,637,185	3,970,178	0.94	1.03	1.03

Table 4.4: Non-contacting pair marginal counts: partner's same attribute. $[S_1S_2]$ shows a strong dependency between secondary structure types of non-contacting residues. ($[A_1A_2]$: $G^2 = 1,329$ with 4 degrees of freedom, $[E_1E_2]$: $G^2 = 4,993$ with 1 degree of freedom, $[S_1S_2]$: $G^2 = 306,659$ with 2 degrees of freedom).

Chapter 3 (Table 3.6), and they do.

Three of the marginal count tables summarize the association between a residue's attribute and the same attribute of the residue's partner (Table 4.4). For example, one table, $[A_1A_2]$ describes the co-occurrence of amino acid types of the two paired residues. I expected that these attributes would be independent.

While all the tables show significant nonrandom association by the G^2 test, this association is orders of magnitude higher for the association between secondary structures than for the association between amino acid types or between solvent accessibilities. Because I am expecting that there should be no significant association between amino acid type or solvent exposure of non-contacting residues, I interpret this data as giving a significance baseline for the G^2 values: $G^2 = 1329$ with four degrees of freedom is not significant in this application; $G^2 = 306,659$ with two degrees of freedom is. There is a very high number of counts, and this results in an unexpectedly large G^2 value for all tables.

Why is there signficant association between secondary structure types of non-contacting residues? There is a clear tendency for alpha to associate with alpha, and beta with beta. Alpha and beta prefer not to be associated. The answer is that there are many proteins which have alpha structure or beta structure but not both,

$[A_1E_2]$	buried2	exposed2	buried2	exposed2
hydrophobic1	2,788,861	2,559,843	0.99	1.01
neutral1	2,828,106	2,563,397	1.00	1.00
polar1	2,763,181	2,476,294	1.01	0.99

$[A_1S_2]$	alpha2	beta2	coil2	alpha2	beta2	coil2
hydrophobic1	1,637,813	1,073,760	2,637,131	1.00	0.99	1.00
neutral1	1,613,664	1,117,063	2,660,776	0.98	1.02	1.00
polar1	1,632,557	1,049,328	2,557,590	1.02	0.99	0.99

$[S_1E_2]$	buried2	exposed2	buried2	exposed2
alpha2	2,605,812	2,278,222	1.02	0.98
beta2	1,650,175	1,589,976	0.97	1.03
coil2	4,124,161	3,731,336	1.00	1.00

Table 4.5: Non-contacting pair marginal counts and ratios of observed to expected counts: partner's different attribute. ($[A_1E_2]$: $G^2 = 379$ with 2 degrees of freedom, $[A_1S_2]$: $G^2 = 2,298$ with 4 degrees of freedom, $[S_1E_2]$: $G^2 = 4,611$ with 2 degrees of freedom).

	A_1	E_1	S_1	A_2	E_2	S_2
A_1		2,455,692	790,047	1,329	379	2,298
E_1			894,805		4,993	4,611
S_1						306,659

Table 4.6: Summary of G^2 values for the nine two-dimensional tables.

or perhaps predominantly one type of secondary structure. In these proteins, there will be no pairs, regardless of sidechain contact, which are alpha/beta combinations.

The remaining three two-dimensional marginal count tables describe association between a residue's attribute and a different attribute of the residue's partner in a non-contacting pair. Using the G^2 significance values determined from Table 4.5, these counts do not show significant association between different attributes of the partner.

The G^2 values for the nine tables are summarized in Table 4.6. Whether or not association exists is indicated more reliably by the relative sizes of the G^2 values, not their absolute values.

Model	Added terms	G^2	df	ΔG^2	% ΔG^2	Δdf
M	$[A_1][A_2][E_1][E_2][S_1][S_2]$	8,137,425	313			
A	$[A_1E_1][E_2A_2]$	3,226,037	309	4911388	61.8	4
B	$[E_1S_1][E_2S_2]$	1,436,426	305	1789611	22.5	4
C	$[A_1S_1][A_2S_2]$	500,139	297	936287	11.8	8
D	$[S_1S_2]$	193,481	293	306658	3.9	4
E	$[E_1E_2]$	189,082	289	4399	0.1	4
F	$[A_1A_2]$	187,347	288	1735	0.0	1
total				7950078	100.0	25

Table 4.7: A hierarchy of models testing pairwise independence of non-contacting residue pairs. At each step, the two-dimensional margin resulting in the largest ΔG^2 was added.

Full six-dimensional contingency table analysis

An analysis using the full six-dimensional contingency table of non-contacting pairs confirms the observations made from the two-dimensional margin tables.

The nomenclature of the models is as follows. Margins are represented in the model names by enclosing their attribute names in brackets. "$[A_1E_1S_1]$" in the model name indicates that margin $[A_1E_1S_1]$ and all its subsidiary margins ($[A_1E_1]$, $[A_1S_1]$, $[E_1S_1]$, $[A_1]$, $[E_1]$, and $[S_1]$) are in the model. I refer to the independence model, $[A_1][A_2][E_1][E_2][S_1][S_2]$, as M. Some models are named relative to M; in other words, I call them "M+[.]", where the additional margins are listed after the plus sign.

Table 4.7 shows a series of models of the six-dimensional contingency table. These models are hierarchical: they were chosen by adding, from one to another in the series, the two-dimensional margin that explains the most difference between the expected and observed data. Only the six margins corresponding to singleton and same-attribute correlations were considered. By comparing each model to the previous one, we get an idea of the strength of association between the two variables in the added margin in the context of the associations already made. For each model, I report the likelihood ratio test statistic G^2, the number of degrees of freedom ("df"), the change ΔG^2 from one model to the next, the percent change $\%\Delta G^2$ over the whole series, and the change Δdf in degrees of freedom.

The singleton terms ([AE], [ES], and [AS] for a single residue) have the greatest

effect. The interactions between residues are significant according to the likelihood ratio statistical test, but (with the exception of $[S_1S_2]$) explain four orders of magnitude less of the variance than same-residue terms. Thus I consider the inter-residue interactions to be negligible, except for $[S_1S_2]$. Clearly, there is some correlation between the secondary structures of non-contacting residues.

To explain the nonrandom association between secondary structures of non-contacting residues, consider the ratios between the observed $[S_1S_2]$ marginal counts and the expected countsshown in Table 4.4. Note that expected counts for coil are about the same as the observed counts. However, these ratios clearly show that there are fewer alpha-beta pairs and more alpha-alpha and beta-beta pairs than expected in the database. This is just what we'd expect given that some proteins are all-alpha or all-beta.

The model $[A_1A_2E_1E_2S_1][A_2A_2E_1E_2S_1]$ corresponds to the expected values used in some pseudopotentials to compute the pairwise preferences. The only residue attributes that it assumes are independent are the amino acid types for the two residues (and higher-order margins involving both A_1 and A_2). The G^2 statistic for the expected table of counts computed from model $[A_1A_2E_1E_2S_1][A_2A_2E_1E_2S_1]$ is 4844, with 144 degrees of freedom. Compared to the observed table of counts, this is statistically significant, though it is four orders of magnitude less than the G^2 model M (one-dimensional margins). I do not expect any correlation for residues which are not in contact. That there exists apparent numerical significance should act as a caution in interpreting the numbers for pairwise correlation of contacting residues!

4.3.2 Contacting pairs; residues grouped into three classes

I now turn to the analysis of pairs of residues that are in contact in the three-dimensional protein structures.

Pairwise dependencies among residue attributes

In this section, I look at pairwise dependencies among residue attributes. This corresponds to the analysis of the nine two-dimensional tables of marginal totals that

Model	G^2	df	$G^2_M - G^2$	$df_M - df$
M	238,712	313	0	0
Environment				
$M+[S_1S_2]$	166,387	309	72,324	4
$M+[E_1E_2]$	205,607	312	33,105	1
$M+[E_1S_1][E_2S_2]$	212,521	309	26,191	4
$M+[E_1S_2][E_2S_1]$	228,113	309	10,599	4
Singleton				
$M+[A_1E_1][A_2E_2]$	165,004	309	73,708	4
$M+[A_1S_1][A_2S_2]$	210,931	305	27,781	8
Pseudo-singleton				
$M+[A_1E_2][A_2E_1]$	217,457	309	21,255	4
$M+[A_1S_2][A_2S_1]$	225,349	305	13,363	8
Pairwise				
$M+[A_1A_2]$	228,071	309	10,641	4

Table 4.8: Models of pairwise dependence for amino acids grouped by hydrophobicity. For each model, the G^2 and df numbers are compared with those of the independence model M.

I performed for the non-contacting pairs in Section 4.3.1. Here, however, I will do a similar analysis using loglinear models built on the full contingency table. The idea is to have the independence model as a reference, and then to add to it two-dimensional margins corresponding to each pairwise dependency. This way we can see the apparent interaction of each pair of attributes in the absence of other higher-order interactions.

The models built for contacting pairs are listed in table 4.8. For each model, I give the likelihood ratio test statistic G^2, and the number of degrees of freedom in the model. In addition, the difference in G^2 between each model and the first model, M, is given. The meaning of the comparison between each model and M is listed in the right-hand column. The models are grouped according to the types of interaction that they add to the pairwise independence model (M).

By looking at $G^2_M - G^2$, and taking into consideration the difference in the number of degrees of freedom between each model and model M, we can get a glimpse of the strength of the association between residue attributes in the pair. As with the non-

Name	Model
BASE	$[A_1][A_2][E_1E_2S_1S_2]$
SING	$[E_1E_2S_1S_2][A_1E_1S_1][A_2E_2S_2]$
PSING	$[A_1E_1E_2S_1S_2][A_2E_1E_2S_1S_2]$
FULL	$[A_1A_2E_1E_2S_1S_2]$

Table 4.9: Names of models corresponding to pseudopotential functions.

Model	G^2	df	ΔG^2	% ΔG^2	Δdf
BASE	104,509	284			
SING	11,703	264	92,806	88.8	20
PSING	3,728	144	7,975	7.6	120
FULL	0	0	3,728	3.6	144
total			104,509	100.0	284

Table 4.10: Models related to the threading score functions. Computed by grouping amino acids into three groups (hydrophobic, neutral, polar).

contacting pairs, there is strong association between attributes of a single residue. There is also strong association between secondary structure types, reflecting the tendency for many proteins to contain predominantly helix or strand. In contacting pairs, there also appears to be significant dependency between the environments of the residues. There is also significant interaction between amino acid type of one residue and the environment of the other residue. And finally, there is some significant pairwise interaction between the amino acid types themselves.

Some of the pairwise dependence between variables might also be explained by other interactions. Given my interest in threading pseudopotentials, I built hierarchies of models to correspond to the parts of a typical pseudopotential and analyzed the incremental reduction in G^2 at each stage in order to determine the relative importance of each part.

Model hierarchy corresponding to pseudopotentials

In this section I describe a model hierarchy built to correspond to the singleton and pairwise terms in a pseudopotential function. Table 4.9 lists the names of the models, and their marginal components. Table 4.10 contains a model series built from the six-way contingency table, in which amino acid type is one of three groups.

Model BASE is the base case. It includes all interactions between structure environment attributes (secondary structure and solvent exposure) for the two contacting residues. It assumes that amino acid type is independent, both between residues, and from the environment.

The next model in the hierarchy is SING. This model incorporates the singleton margins $[A_1E_1S_1]$, $[A_2E_2S_2]$, and all subsets of these. These correspond to the singleton terms in a pseudopotential. The ratio of expected counts from model SING to those from model BASE is closely related to the ratio of observed singleton counts to singleton counts predicted by the assumption of random association. This last ratio is the one used to define the pseudopotentials. These singleton terms account for 88.8% of the G^2 statistic.

Next is model PSING. At this stage we are adding five-dimensional margins $[A_1E_1E_2S_1S_2]$, $[A_2E_1E_2S_1S_2]$ and margins corresponding to all subsets thereof, most notably $[A_1E_2S_2]$ and $[A_2E_1S_1]$. These represent the interactions of an amino acid type with the environment of the residue with which it is in contact. Thus adding these terms corresponds to pseudopotential terms representing the association of amino acid type and partner's secondary structure and solvent exposure. These terms account for 7.6% of the G^2 statistic, for this case of grouping amino acids into three types.

Most pseudopotentials that consider information about residue pairs do not take into account the association of an amino acid with its neighbor's structure type. On the other hand, Ouzonis and colleagues consider the neighbor's secondary structure, but not the amino acid type [Ouzounis *et al.*, 1993].

Finally, the full model corresponds to the observed data. Comparing this model to the previous model, the added terms are those corresponding to pairwise interactions of the amino acid types. Only 3.6% of the G^2 statistic is explained by these interactions. The pairwise potential function used in several threading methods is computed by taking the ratio of the observed counts (full model) to those predicted by model PSING, although researchers generally don't group the amino acids.

Model	G^2	df	$G^2_M - G^2$	$df_M - df$
M	304,998	14,355	0	0
Environment				
M+[S_1S_2]	232,673	14,351	72,325	4
M+[E_1E_2]	271,893	14,354	33,105	1
M+[E_1S_1][E_2S_2]	278,806	14,,351	26,192	4
M+[E_1S_2][E_2S_1]	294,399	14351	10,599	4
Singleton				
M+[A_1E_1][A_2E_2]	217,419	14,317	87,580	38
M+[A_1S_1][A_2S_2]	254,655	14,279	50,343	76
Pseudo-singleton				
M+[A_1E_2][A_2E_1]	280,085	14,317	24,913	38
M+[A_1S_2][A_2S_1]	284,866	14,279	20,133	76
Pairwise				
M+[A_1A_2]	283,969	13,994	21,029	361

Table 4.11: Models of pairwise dependence for 20 amino acid types. All nine two-dimensional margins are added to the independence model M.

4.3.3 Contacting residues; all twenty amino acids

Now I turn to the case where the amino acid attribute of each residue has 20 categories.

Pairwise dependencies between residue attributes

Table 4.11 shows models of pairwise dependence between residue attributes. Again we see significant pairwise relationships between all nine pairs of attributes.

Model hierarchy corresponding to pseudopotentials

In this section I build a model hierarchy identical to that built in the previous section, but for all 20 amino acids separated into their own classes.

The singleton terms explain 75.8% of the G^2 statistic; amino acid type and partner's environment explain 8.6%; and pairwise amino acid type terms explain 15.7%.

I compared this model hierarchy to one generated by a different, smaller data set of 49 proteins. The results are shown in Table 4.13. For the smaller data set, the pairwise terms appear to account for about half of the information as measured

Model	G^2	df	ΔG^2	% G^2	Δdf
BASE	170,795	14,326			
SING	41,353	14,136	129,442	75.8	190
PSING	26,737	12,996	14,616	8.6	1140
FULL	0	0	26,737	15.7	12,996
total			170,795	100.0	14,326

Table 4.12: Models related to the threading score function. Computed with 20 amino acid types.

by ΔG^2. This is consistent with the results of Bryant and Lawrence [Bryant and Lawrence, 1993]. Bryant and Lawrence use 161 proteins in their loglinear analysis. They classify the residues by their amino acid type (20 classes) and the distance between the side-chains (six classes). They find that singleton terms account for about two-thirds of the apparent specificity in pairwise residue interactions, with pairwise terms accounting for the remaining one-third. They do not directly consider structural parameters such as solvent exposure and secondary structure, but they do consider the distance between residues in a pair.

The relative importance of the pairwise appears much larger in the smaller data set. This might shed light on one problem with estimates of the relative importance of pairwise and singleton terms: it is important to use enough data, and/or a simple enough structure representation.

There are indications in the inverse folding approach to protein structure prediction that single-residue terms are doing most of the work in identifying correct matches between sequence and structure. Interestingly, much work has been put into developing threading algorithms that can handle pairwise and higher interactions. It is possible that the inability of pairwise interactions to improve threading results might reflect the overriding singleton effect of hydrophobic collapse. Hydrophobic collapse, in which local sequence hydrophobicity is the major player, is the driving force for defining the overall fold of the protein. Interactions between specific residues would then be responsible for fixing the configuration, but not making gross changes in topology.

248 proteins			
	3 groups	20 groups	20 groups padded
SING	88.8	75.8	72.7
PSING	7.6	8.6	12.6
FULL	3.6	15.7	14.7

49 proteins			
	3 groups	20 groups	20 groups padded
SING	89.5	58.0	49.8
PSING	7.8	9.2	22.8
FULL	2.6	32.8	27.4

Table 4.13: $\%\Delta G^2$ relative to the BASE model for two protein sets.

On the other hand, it could be that other aspects of the threading methods are responsible for the failure to take advantage of structure-determining pairwise interactions. Either the structure representation or the pseudopotential functions could be at fault. My results indicate that low sample size could be part of the problem.

To ameliorate low sample size problems, it is common practice to pad the observed counts in some way. One option is to add a constant offset to each cell (see also Section 8.1.3). I did this to my table of counts and built the same hierarchy of models; the results are summarized in Table 4.13. For the smaller protein set, there is a dramatic increase in the importance of the terms relating amino acid type to the neighbor's environment.

4.4 Conclusions

Only 15.7% of the nonrandom association of pairs (assuming a complete description of the pair environment) is explained by correlation between amino acids. Clearly singleton terms are much more important than pairwise terms. This explains the failure of many threading pseudo-potential functions to show much improvement when pairwise terms are added to the singleton terms.

8.6% of the nonrandom association of pairs can be explained by association of amino acid type with the other residue's environment. In a sense this is a "pseudo-singleton" term; the environments of the partners are constant in threading, a function of the structural model. This suggests that some improvement in performance might be obtained by expanding the singleton environment description to incorporate the environments of neighboring positions. An optimal solution to the threading problem could be found with dynamic programming, because we're dealing with singleton terms only.

Sample size is a problem when looking at pairwise data, especially with a representation of a complexity that I use (20 amino acid types, three secondary structure types, and two solvent exposure types). The more proteins that are used in deriving the counts, the better.

From the non-contacting pairs results, it appears that statistical tests about association between residue attributes must be taken with a grain of salt; using information about the relative sizes of measures of fit is a more meaningful approach.

Chapter 5

Distinguishing Parallel from Antiparallel

5.1 Summary

In 1979 Lifson and Sander wrote [Lifson and Sander, 1979],

> "As more protein structure data become available, further distinctions of secondary structure elements according to the type of tertiary contacts should be made. For example, one can distinguish different hydrogen-bonding positions in beta-sheets, solvent-exposed and interior faces of sheets or helices, segments in tertiary contact with sheets compared with those in contact with helices. Such distinctions are likely to lead to more clearcut statistical preferences, and also serve as as starting point for predicting tertiary structure."

In this chapter I consider a finer classification than alpha/beta/coil by dividing the beta class into parallel and antiparallel. I have evidence that this division results in a more useful representation: Lifson and Sander found that the amino acid compositions of parallel and antiparallel beta sheets were quite different [Lifson and Sander, 1979]. This observation led to the suggestion that parallel and antiparallel conformations be distinguished in structure prediction methods. Here, I begin by updating the Lifson

and Sander analysis for a larger set of proteins. I find qualitatively similar results, including a difference between the amino acid composition of parallel and antiparallel beta sheets.

However, I then ask whether this difference in composition might be due to a difference in solvent exposure of parallel and antiparallel beta sheets. I observe that the compositions of buried beta strands is different than that of exposed beta strands. The difference between parallel and antiparallel structure remains strong when looking only at buried structure; in particular, the beta-branched residues Val and Ile show some preference for parallel over antiparallel beta structure. There are also differences between buried and exposed parallel beta strands, and between buried and exposed antiparallel beta strands.

Looking at frequencies of occurrence and their ratios does not tell us the relative importance of strand direction and solvent exposure, nor the degree to which they are dependent. To answer these questions, and to handle the effects of different total counts of parallel, antiparallel, exposed, and buried residues, I turn to contingency table analysis using loglinear models. I consider a coarser classification of amino acid, into three groups. Such a classification guarantees enough counts for statistical significance analysis, and allows finer divisions of other local structure descriptors. I find that the partitioning of hydrophobic residues to the interior is the most dominant effect. Moreover, the tendency for parallel sheets to be buried explains much more variance than the partitioning of residue types between different parallel and antiparallel beta sheet structure.

To support this work, I have written a computer program that computes beta pair counts and statistics for any set of proteins. This program is potentially useful for structure prediction algorithms employing empirically-derived parameters, particularly for cross-validation of these methods.

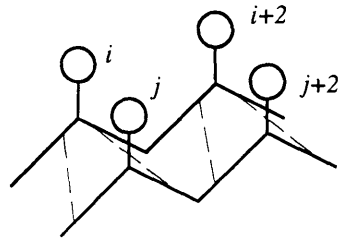


Figure 5-1: Schematic diagram of topological relationships in parallel strands. Lines are draw connecting C_α atoms. Circles represent side chains. Dashed lines represent hydrogen bonds. Residues i and j are beta pairs (β); residues i and $j+2$ are diagonal pairs (δ); residues i and $i+2$ are γ-structure.

5.2 Methods

5.2.1 Protein data

I used the Kabsch and Sander DSSP files for secondary structure and solvent accessibility information [Kabsch and Sander, 1983].

In the sections reviewing the Lifson and Sander results, 102 proteins from the Brookhaven database were used, as listed in Appendix B (subunits are indicated where appropriate as the last letter of the name). These are the proteins used by Jones and colleagues in their work on threading [Jones *et al.*, 1992], and are nonhomologous and well-refined.

In the contingency table analysis section, I used proteins from the pdb_select list distributed by EMBL [Hobohm *et al.*, 1992]. There are 252 proteins in this list for which DSSP and PDB files are available.

5.2.2 Definition of secondary structure, and topological relationships.

Secondary structure is as defined by Kabsch and Sander, for their DSSP program [Kabsch and Sander, 1983].

I am particularly interested in three same-sheet topological relationships between core element positions, illustrated in figure 5-1.

1. Beta pairs. My definition of a beta pair is the two central residues in a bridge (for

a definition of bridge, see Section 2.1.1). I refer to this structure throughout this thesis as β-structure. Antiparallel beta pairs are designated by β_A, and parallel beta pairs by β_P. A beta pair is similar to what Lifson and Sander [Lifson and Sander, 1980] refer to as a "residue contact." For comparison, their three-part definition of two residues on adjacent beta strands that are in contact is:

- The alpha carbons must be less than 7 angstroms apart.
- The Ca-Cb vectors must not be more than 90 degrees different; this ensures that the side chains are on the same side of the sheet.
- The hydrogen bond donors and acceptors must be either both pointing at each other or both pointing away from each other; this requirement selects for the canonical beta-structure hydrogen bonding pattern.

2. Diagonal pairs (δ-structure).

   If (i, j) and $(i+2, j+2)$ are both β_P pairs, then $(i, j+2)$ and $(i+2, j)$ are diagonal pairs (denoted by δ_P). If (i, j) and $(i+2, j-2)$ are β_A, then $(i, j-2)$ and $(i+2, j)$ are δ_A.

3. i, $i+2$ pairs (γ-structure). Residues i and $i+2$ are in a beta strand.

5.2.3 Counting and statistics.

Following Lifson and Sander [Lifson and Sander, 1979], I count pair members rather than residues in deriving the single residue frequencies. Thus residues on the edge of a sheet (participating in only one beta pair) are counted once, and residues in the interior of a sheet (participating in two beta pairs) are counted twice.

For each of the three topological relationships of interest, I count N_{ij}, the number of occurrences of each pair of residues i, j. I do not distinguish here between the two residues in a pair, and so the counts are symmetric ($N_{ij} = N_{ji}$).

Counts may be filtered in several ways.

- *Strand pair direction.* Beta pairs and diagonal pairs are either antiparallel or parallel. This is not relevant for i, $i+2$ pairs, as they are on the same strand.

- *Surface accessibility*, as defined by the DSSP program [Kabsch and Sander, 1983], may be used to consider only residues whose surface accessibility is more or less than a given threshold (30% of maximum accessibility in this chapter). I use the superscript $^{\text{buried}}$ to indicate a count or frequency using only buried residues. For example, β_A^{buried} denotes antiparallel buried beta pair structure.

- *Sheet position.* Beta-sheet positions are either "edge" (participating in only one beta pair) or "interior" (participating in two beta pairs). β^{edge} and β^{interior} correspond to edge and interior structure.

- *Contact.* Residue pairs are either in contact or not (for definition of contact, see below). β^{touching} represents beta pairs in contact.

I count N_{ij}^{β}, the number of beta pairs composed of residues of type i and type j. N_{ij}^{δ} and N_{ij}^{γ} are similarly defined. The total number of beta pair contacts made by residues of type i is

$$N_i^{\beta} = N_{ii}^{\beta} + \sum_j N_{ij}^{\beta}.$$

The total number of beta pair contacts is

$$N^{\beta} = \sum_i N_i^{\beta}.$$

Lifson and Sander compute the conformational preference of a given amino acid residue as a ratio of its frequency in a beta structure to its global frequency. From the beta pair contact counts, I compute conformational preference of residue type i for a structure, say antiparallel sheet β_{A}, as

$$C_i = \frac{f_i^{\beta_A}}{f_i^{\text{global}}},$$

where f_i^{global} is the ratio of the number of occurrences of residue type i in the database to the total number of residues in the database. Note that Lifson and Sander use a frequency computed by Feldman [Feldman, 1976] for the global frequency, whereas I compute global frequency directly from my test set of proteins.

Lifson and Sander use the conformational preferences to define classes of amino acids based on their beta sheet propensities. A residue i is defined as a "sheet makers" if $C_i > 1.5$. Sheet breakers have $C_i < 1/1.5$.

5.2.4 Contingency table analysis

A contingency table analysis using loglinear models was performed to determine the relative importance of amino acid group, strand direction, and solvent accessibility. This approach looks at the difference in error (as measured by the likelihood test statistic G^2, which has a χ^2 distribution) between nested models, to determine how much of the error is accounted for by which terms. Each pair of models differs by the inclusion or exclusion of one marginal term. Each marginal term corresponds to a set of variables; the absence of a term in a model indicates an assumption of independence among those variables. Thus loglinear analysis tests for the presence and the strength of conditional dependence among variables in the model, as well as the power of each of the variables and combinations thereof to explain the observed data.

5.3 Results and Discussion

5.3.1 Amino acid compositions of parallel and antiparallel beta structure

The total number of residues, beta residue, and beta pairs is shown in table 5.1.

Beta residues make up 3967/19789 = 20% of the residues in the data set. The Kabsch and Sander definition of beta residue is fairly stringent compared to other definitions. There are more singly (sheet-edge) paired residues than doubly paired residues. There are some beta residues that are not beta-paired; these are residues in the middle of strands that are part of the strand but not paired. Most of them are beta bulges.

I repeated the Lifson and Sander work, using a different (and larger) set of proteins

Total residues in training set:	19789
Beta residues in training set:	3967
Beta pairs in training set:	2580
Doubly beta-paired beta residues:	1349
Singly beta-paired beta residues:	2462
Non beta-paired beta residues:	156
Parallel beta pairs:	802
Antipar. beta pairs:	1724
Parallel diagonal beta pairs:	752
Antipar. diagonal beta pairs:	1783

Table 5.1: Summary of beta pair counts.

and a different definition of beta strand. My results are qualitatively similar to theirs. Table 5.2 lists the counts and frequencies (count for an amino acid divided by the total number of counts) for all residues, and those found in parallel, antiparallel, and any beta structure. I find, as do Lifson and Sander, that Val, Ile and Leu dominate beta structure, particularly in parallel sheets. I find approximately twice as many antiparallel beta pairs as parallel ones; Lifson and Sander found a ratio of three to one.

The frequencies that differ from those found by Lifson and Sander by more than one percentage point are listed in Table 5.3. I find Ile and Leu more frequently in parallel sheet structures than do Lifson and Sander; this is probably due to my more stringent definition of beta sheet structure. I tend to leave out some of the residues at the periphery of the Lifson and Sander sheets. This also explains the apparent decrease in occurrence of Ala, Ser, and Thr in antiparallel sheets.

Can we make sense of the observed frequencies of occurrence? Chothia and Janin [Chothia and Janin, 1982] ascribe the prevalence of Val, Ile and Leu in β structure to the fact that their branched side chains are capable of forming a "smooth, well-packed surface on the β sheets." Finkelstein and Nakamura [Finkelstein and Nakamura, 1993] suggest that aromatic residues are required in antiparallel beta sheets to occupy intrinsic cavities that cannot be filled by aliphatic residues. This may explain why Tyr and Trp have greater frequencies in antiparallel structure, while their parallel frequencies are similar to their global frequencies. Phe has higher frequency in an-

Counts and Frequencies of beta-paired residues								
	Whole dbase		Parallel		Antiparallel		Both	
AA	N_i	f_i	$N_i^{\beta_P}$	$f_i^{\beta_P}$	$N_i^{\beta_A}$	$f_i^{\beta_A}$	N_i^{β}	f_i^{β}
G	1674	0.085	81	0.050	172	0.050	253	0.050
P	895	0.045	12	0.007	52	0.015	64	0.013
D	1211	0.061	33	0.021	86	0.025	119	0.024
E	1172	0.059	40	0.025	123	0.036	163	0.032
A	1772	0.090	112	0.070	210	0.061	322	0.064
N	876	0.044	34	0.021	90	0.026	124	0.025
Q	702	0.035	19	0.012	112	0.032	131	0.026
S	1316	0.067	69	0.043	216	0.063	285	0.056
T	1126	0.057	87	0.054	264	0.077	351	0.069
K	1345	0.068	52	0.032	185	0.054	237	0.047
R	774	0.039	38	0.024	133	0.039	171	0.034
H	458	0.023	26	0.016	84	0.024	110	0.022
V	1418	0.072	328	0.204	437	0.127	765	0.151
I	1032	0.052	256	0.160	276	0.080	532	0.105
M	377	0.019	43	0.027	71	0.021	114	0.023
C	377	0.019	22	0.014	106	0.031	128	0.025
L	1594	0.081	194	0.121	321	0.093	515	0.102
F	754	0.038	82	0.051	212	0.061	294	0.058
Y	667	0.034	54	0.034	222	0.064	276	0.055
W	249	0.013	22	0.014	76	0.022	98	0.019
Total	19789	1.000	1604	1.000	3448	1.000	5052	1.000

Table 5.2: Beta paired residue counts and frequencies.

	Residue	here	LS79	diff
Antiparallel	Ala	6.1	7.6	-1.5
	Ser	6.3	7.8	-1.5
	Thr	7.7	8.8	-1.1
	Phe	6.1	4.3	1.8
Parallel	Gly	5.0	7.2	-2.2
	Ala	7.0	8.4	-1.4
	Ser	4.3	6.3	-2.0
	Thr	5.4	4.0	1.4
	Ile	16.0	12.0	4.0
	Leu	12.1	10.1	2.0
All-beta	Ala	6.4	7.8	-1.4
	Ser	5.6	7.4	-1.8
	Ile	10.5	8.3	2.2
	Phe	5.8	4.4	1.4

Table 5.3: Residues whose f_i^β as computed here and by Lifson and Sander differ by more than one percentage point. Frequencies are given in percent.

tiparallel than parallel structure, and the parallel frequency is higher than the global frequency. Pro is rare in beta sheet structure; the backbone conformation it imposes is not consistent with beta sheet. Charged and polar amino acids appear somewhat less frequently than on average, except that Thr appears quite frequently in antiparallel structure, which may be partly due to the fact that it is branched at the beta carbon like Val and Ile, and therefore may help shape the beta sheet. Gly and Ala occur less often in beta sheet than in other structures. Both are small, and the hydrophobicity of their side chains are dominated by the hydrophilicity of the backbones.

Table 5.4 shows conformational preferences for the amino acids, for the β class, and compares them to those determined by other researchers [Lifson and Sander, 1979, Chou and Fasman, 1974, Garnier *et al.*, 1978, Levitt, 1978].

Comparing my results (column 1) with those of Lifson and Sander (column 3), we see that there is agreement as to which residues prefer beta conformation and which do not. In addition, the ordering of amino acids is similar, and exact within the following groups: $P < D < ENG < K < AQSRH < MTLCF < WY < IV$.

If I separate the conformational preferences of β_P and β_A (Table 5.5), I find a similar qualitative agreement with the following notable exceptions: Lifson and Sander

AA	1	2	3	4	5	6
G	0.59	0.64	0.61	0.75	0.66	0.92
P	0.28	0.30	0.40	0.55	0.84	0.64
D	0.38	0.63	0.48	0.54	0.64	0.72
E	0.54	0.74	0.61	0.37	0.61	0.75
A	0.71	0.68	0.92	0.83	0.79	0.90
N	0.55	0.69	0.60	0.89	0.66	0.76
Q	0.73	0.76	0.95	1.10	1.13	0.80
S	0.85	0.78	0.82	0.75	0.84	0.95
T	1.22	0.92	1.12	1.19	1.14	1.21
K	0.69	1.13	0.70	0.74	0.72	0.77
R	0.87	1.12	0.93	0.93	1.04	0.99
H	0.94	0.87	0.93	0.87	0.78	1.08
V	2.11	1.51	1.81	1.70	1.97	1.49
I	2.02	1.39	1.81	1.60	1.95	1.45
M	1.18	1.00	1.19	1.05	1.26	0.97
C	1.33	1.06	1.16	1.19	1.55	0.74
L	1.27	0.96	1.30	1.30	1.26	1.02
F	1.53	1.23	1.25	1.38	1.30	1.32
Y	1.62	1.47	1.53	1.47	1.49	1.25
W	1.54	1.33	1.54	1.37	0.90	1.14

Table 5.4: Conformational preferences for all-beta as computed here and in four references. Column 1: computed here. Column 2: computed here; buried residues only. Column 3: Lifson and Sander, 1979. Column 4: Chou and Fasman, 1974. Column 5: Garnier *et al.*, 1978. Column 6: Levitt, 1978.

Beta Conformation Preferences								
	Parallel		Antiparallel		Par./Anti.		All beta	
AA	here	LS79	here	LS79	here	LS79	here	LS79
G	0.60	0.79	0.59	0.56	1.01	1.41	0.59	0.61
P	0.17	0.35	0.33	0.42	0.50	0.83	0.28	0.40
D	0.34	0.50	0.41	0.47	0.82	1.08	0.38	0.48
E	0.42	0.59	0.60	0.62	0.70	0.96	0.54	0.61
A	0.78	1.00	0.68	0.90	1.15	1.11	0.71	0.92
N	0.48	0.54	0.59	0.62	0.81	0.88	0.55	0.60
Q	0.33	0.28	0.92	1.18	0.36	0.24	0.73	0.95
S	0.65	0.70	0.94	0.87	0.69	0.80	0.85	0.82
T	0.95	0.59	1.35	1.30	0.71	0.45	1.22	1.12
K	0.48	0.59	0.79	0.74	0.60	0.79	0.69	0.70
R	0.61	0.68	0.99	1.02	0.61	0.67	0.87	0.93
H	0.70	0.38	1.05	1.12	0.67	0.34	0.94	0.93
V	2.85	2.63	1.77	1.53	1.61	1.72	2.11	1.81
I	3.06	2.60	1.53	1.54	1.99	1.69	2.02	1.81
M	1.41	1.49	1.08	1.09	1.30	1.37	1.18	1.19
C	0.72	0.91	1.61	1.24	0.45	0.73	1.33	1.16
L	1.50	1.42	1.16	1.26	1.30	1.13	1.27	1.30
F	1.34	1.30	1.61	1.23	0.83	1.06	1.53	1.25
Y	1.00	1.08	1.91	1.68	0.52	0.74	1.62	1.53
W	1.09	0.89	1.75	1.75	0.62	0.51	1.54	1.54

Table 5.5: Conformational preferences for parallel and antiparallel sheet as computed here and in Lifson and Sander 79 (LS79). Also shown are the conformational preference ratio P/A and the conformational preferences for all beta sheet residues.

class	LS 79	here
A+P+	VIMLFY	VIMLFYW
A+P-	QTRHCW	THC
A-P-	GPDEANSK	GPDEANQSKR
SM A	W>Y>V>I	Y>V>W>C>I
SM P	V>I>M	I>V>L
SB A	P<D<G<E<N	P<D<NG<E
SB P	Q<P<H<D<E<T<K	P<Q<D<E<KN<G<R<S

Table 5.6: Conformational classification of residues. A+P+: favorable in both β_A and β_P (preference > 1.0 in both); A+P−: favorable in β_A but unfavorable in β_P ; A−P−: unfavorable in both β_A and β_P ; SM A: Best sheet makers in β_A (preference > 1.5 in both β_A and β_P); SM P: best sheet makers in β_P ; SB A: worst sheet breakers in β_A (preference $\leq 1/1.5$); SB P: worst sheet breakers in β_P .

calculate the conformational preference of Trp to be unfavorable at 0.89 in β_P; I calculate a favorable 1.09. Lifson and Sander calculate the conformational preference of Gln to be 1.18 in β_A; I calculate 0.92.

Finally, Lifson and Sander consider the "conformational classification" of beta sheet residues, which are based on the conformational preferences. This classification can be useful in structure prediction algorithms; the Chou-Fasman algorithm uses similar classes for nucleation and termination of secondary structures [Chou and Fasman, 1978]. Classes include: favorable in both β_P and β_A, favorable in β_A but unfavorable in β_P, unfavorable in both, best sheet "makers," and worst sheet "breakers." My results (Table 5.6) show minor differences in this conformational classification. For example, I find Trp to be favorable in both β_A and β_P, whereas Lifson and Sander find Trp favorable in β_A but not in β_P. I find Phe to be a good β_A sheet maker, but Lifson and Sander do not.

In summary, I show general agreement with the results of Lifson and Sander. I find that the orderings which they cite within their conformational classification are not reliable; changing the protein set (results not shown) can result in markedly different orderings. In the protein sets of both Lifson and Sander, and Jones, I consistently find the following residue classifications (for class definition see the caption for Table 5.6). Val, Ile, Met, Leu, Phe and Tyr are favorable in both β_A and β_P. Thr, His and Cys are favorable in β_A but not in β_P. Tyr and Val are antiparallel sheet makers. Ile and

AA	All	Buried	Difference	Classification
L	8.1	13.5	+5.4	hydrophobic
I	5.2	10.4	+5.2	hydrophobic
V	7.2	13.1	+4.9	hydrophobic
A	9.0	12.6	+3.6	neutral
F	3.8	6.0	+2.2	hydrophobic
G	8.5	10.6	+2.1	neutral
C	1.9	3.7	+1.8	hydrophobic
M	1.9	3.0	+1.1	hydrophobic
W	1.3	1.5	+0.2	hydrophobic
H	2.3	1.7	-0.6	polar
Y	3.4	2.6	-0.8	hydrophobic
T	5.7	4.9	-0.8	neutral
S	6.7	5.7	-1.0	neutral
P	4.5	2.6	-1.9	neutral
N	4.4	2.2	-2.2	polar
Q	3.5	1.2	-2.3	polar
R	3.9	0.7	-3.2	polar
D	6.1	2.2	-3.9	polar
E	5.9	1.2	-4.7	polar
K	6.8	0.4	-6.4	polar

Table 5.7: Comparison of frequencies (in per cent) for all residues and buried residues.

Val are parallel sheet makers. Pro, Asp, Asn, and Gly are antiparallel sheet breakers, and Gln, Pro, Asp and Glu are parallel sheet breakers.

5.3.2 Solvent exposure

Lifson and Sander found a great difference between the amino acid compositions of β_A and β_P. To illustrate this difference, they calculated the ratio of β_P to β_A propensities, and found that Val, Ile, Gly, and Met prefer β_P to β_A, while Gln, Thr, His, Arg, Cys and Trp prefer A to P. Note that Val and Ile (P-preferring) are hydrophobic and branched at the beta carbon. Some of the residues which prefer β_A to β_P are polar; all have a hydrogen-bond donor or acceptor.

However, β_P structure is almost always buried: β_A structure is often amphiphilic. How much of the difference between β_A and β_P propensities found by Lifson and Sander is due to the fact that β_A structure is often partly on the protein surface? I repeated the analysis for buried residues only. My results are shown in Tables 5.7

Res.	Globally		Parallel		Antiparallel		All beta	
G	599	0.106	72	0.063	120	0.071	192	0.068
P	144	0.026	8	0.007	14	0.008	22	0.008
D	123	0.022	11	0.010	28	0.017	39	0.014
E	70	0.012	13	0.011	13	0.008	26	0.009
A	713	0.126	92	0.081	150	0.089	242	0.086
N	125	0.022	18	0.016	25	0.015	43	0.015
Q	68	0.012	3	0.003	23	0.014	26	0.009
S	324	0.057	43	0.038	84	0.050	127	0.045
T	276	0.049	48	0.042	79	0.047	127	0.045
K	23	0.004	6	0.005	7	0.004	13	0.005
R	41	0.007	5	0.004	18	0.011	23	0.008
H	96	0.017	13	0.011	29	0.017	42	0.015
V	739	0.131	269	0.236	289	0.171	558	0.197
I	584	0.104	219	0.192	188	0.111	407	0.144
M	170	0.030	40	0.035	45	0.027	85	0.030
C	209	0.037	20	0.018	91	0.054	111	0.039
L	763	0.135	152	0.133	217	0.128	369	0.130
F	341	0.060	67	0.059	143	0.085	210	0.074
Y	147	0.026	27	0.024	81	0.048	108	0.038
W	87	0.015	13	0.011	45	0.027	58	0.021
Total	5642	1.000	1139	1.000	1689	1.000	2828	1.000

Table 5.8: Counts (N_i^{buried}) and frequencies (f_i^{buried}) of buried residues.

(global frequencies) 5.8 (beta frequencies) and 5.9 (conformational preferences).

Table 5.7 examines the difference between the global frequencies of all and buried residues. The classification labelings are those of von Heijne and Blomberg [von Heijne and Blomberg, 1978]. In general, as we would expect, the hydrophobic residues increase in frequency when considering only buried residues, while the polar residues decrease in frequency. 1139/1604 = 69% of parallel beta-pair partners are buried, as opposed to only 1689/3448 = 47% of antiparallel beta-pair partners. Parallel sheets occur most often buried in the hydrophobic core of proteins. Thus I expect a larger change in antiparallel frequencies when considering only buried residues; in both cases I expect to see an increase in hydrophobic residues.

Compare the buried counts and frequencies in Table 5.8 to those in Table 5.2. I do in fact see a larger change in antiparallel frequencies than in parallel frequencies (as a larger fraction of antiparallel residues are not buried). The standard deviation of the

	All beta pairs				Buried beta pairs			
Res.	Par.	Anti.	Both	Par./Anti.	Par.	Anti.	Both	Par./Anti.
G	0.60	0.59	0.59	1.01	0.60	0.67	0.64	0.89
P	0.17	0.33	0.28	0.50	0.28	0.32	0.30	0.85
D	0.34	0.41	0.38	0.82	0.44	0.76	0.63	0.58
E	0.42	0.60	0.54	0.70	0.92	0.62	0.74	1.48
A	0.78	0.68	0.71	1.15	0.64	0.70	0.68	0.91
N	0.48	0.59	0.55	0.81	0.71	0.67	0.69	1.07
Q	0.33	0.92	0.73	0.36	0.22	1.13	0.76	0.19
S	0.65	0.94	0.85	0.69	0.66	0.87	0.78	0.76
T	0.95	1.35	1.22	0.71	0.86	0.96	0.92	0.90
K	0.48	0.79	0.69	0.60	1.29	1.02	1.13	1.27
R	0.61	0.99	0.87	0.61	0.60	1.47	1.12	0.41
H	0.70	1.05	0.94	0.67	0.67	1.01	0.87	0.66
V	2.85	1.77	2.11	1.61	1.80	1.31	1.51	1.38
I	3.06	1.53	2.02	1.99	1.86	1.08	1.39	1.73
M	1.41	1.08	1.18	1.30	1.17	0.88	1.00	1.32
C	0.72	1.61	1.33	0.45	0.47	1.45	1.06	0.33
L	1.50	1.16	1.27	1.30	0.99	0.95	0.96	1.04
F	1.34	1.61	1.53	0.83	0.97	1.40	1.23	0.69
Y	1.00	1.91	1.62	0.52	0.91	1.84	1.47	0.49
W	1.09	1.75	1.54	0.62	0.74	1.73	1.33	0.43

Table 5.9: Conformational preferences for all beta residues and buried beta residues.

frequency differences across all 20 amino acids from all to buried is 1.5% for parallel and 2.5% for antiparallel. When I consider only buried beta pairs, the compositions of parallel and antiparallel sheets are somewhat more similar: the standard deviation of the frequency differences from parallel to antiparallel is 2.91% for all beta pairs and 2.67% for buried beta pairs. We also see that both antiparallel and parallel hydrophobic residue frequencies increase when only buried residues are considered, except for Cys and Met in parallel sheets, which show a slight decrease in frequency.

The relative frequencies of the amino acids remain the same for the most part in buried beta structure. The four most common beta residues, Val, Ile, Leu and Ala, increase in frequency as we move to buried sheet structure, both for β_P (from 56% to 65%) and β_A (from 36% to 50%). Charged and polar residues, particularly Lys, Arg and Glu, decrease in frequency.

The conformational preferences for buried beta residues are listed as column 2 in

Switch Residues		
Residue	all	buried
Thr	1.22	0.92
Leu	1.27	0.96
Lys	0.69	1.13
Arg	0.87	1.12

Table 5.10: Residues which switch beta propensity going from all beta pairs to buried beta pairs only

Sheet Makers			
	antiparallel	parallel	all-beta
LS 79	W Y V I	V I M	V I Y W
here	Y V W F I	I V	V I Y W F
here, buried only	Y W	I V	V

Table 5.11: Sheet makers in Lifson and Sander 1979; as computed here; and as computed here for buried beta pairs only.

Table 5.4. The residues which switch beta propensity (favorable to unfavorable or vice versa) going from all residues to buried residues only are shown in table 5.10. Lys and Arg don't prefer beta conformation in general; nor do they like to be buried, but if they have to be buried, then they would just as soon be in beta conformation. Thr and Leu are found more often in beta conformation in general, but when buried they don't prefer beta.

One interesting result of looking at only buried residues is the reduction in the number of sheet makers; see table 5.11.

Just considering the preference for parallel over antiparallel, or vice versa, I find that in completely buried structures, some residues remain parallel-preferring (VIM), antiparallel-preferring (DQSRHCFYW), or neutral (GAN); a couple with very low counts switch preference from antiparallel to parallel (EK); a couple switch from antiparallel to neutral (PT); and L switches from parallel-preferring to neutral.

While some of the apparent beta-structure propensities found by Lifson and Sander were artifacts due to the hydrophobic-favoring tendency of parallel sheets, some of the propensities remain the same, and most are qualitatively similar. The preference for Ile and Val in parallel sheets observed by Lifson and Sander is plainly seen here. It is likely that these beta-carbon-branched hydrophobic side chains are

Counts and Frequencies, Sheet Edge and Interior						
AA	edge beta		interior beta		Whole dbase	AA type
	N_i^{edge}	f_i^{edge}	N_i^{int}	f_i^{int}	f_i	
P	65	0.03	0	0.00	0.05	neutral
D	81	0.03	20	0.01	0.06	polar
W	39	0.02	31	0.02	0.01	hydrophobic
H	61	0.02	26	0.02	0.02	polar
N	75	0.03	26	0.02	0.04	polar
Q	81	0.03	27	0.02	0.04	polar
E	109	0.04	29	0.02	0.06	polar
M	42	0.02	37	0.03	0.02	hydrophobic
C	62	0.03	36	0.03	0.02	hydrophobic
R	96	0.04	38	0.03	0.04	polar
K	148	0.06	46	0.03	0.07	polar
G	147	0.06	56	0.04	0.08	neutral
F	125	0.05	87	0.06	0.04	hydrophobic
T	201	0.08	77	0.06	0.06	neutral
Y	100	0.04	93	0.07	0.03	hydrophobic
A	139	0.06	96	0.07	0.09	neutral
I	202	0.08	166	0.12	0.05	hydrophobic
L	219	0.09	157	0.12	0.08	hydrophobic
V	297	0.12	241	0.18	0.07	hydrophobic

Class	Exterior		Interior		Whole database	Class
0	1086	0.44	848	0.63	0.33	hydrophobic
1	725	0.29	289	0.21	0.34	neutral
2	651	0.26	212	0.16	0.33	polar

Table 5.12: Counts and frequencies for sheet interior and exterior.

important in structurally maintaining the sheet. The difference between β_{A} and β_{P} conformational preferences is due to the stricter conformational requirements of parallel sheets.

5.3.3 Position in sheet

I asked whether the amino acid composition of the edges of beta sheets is different than that of the interior of the beta sheets. I considered sheet interior residues only. Results are shown in Table 5.12. Going from exterior to interior, there is a general decrease in the frequencies of polar and neutral residues, and an increase in the frequencies of the hydrophobic residues.

Three-Class Beta Pair Counts and Frequencies									
		All residues		Parallel		Antiparallel		Both	
Class	Residues	N_i	f_i	$N_i^{\beta_P}$	$f_i^{\beta_P}$	$N_i^{\beta_A}$	$f_i^{\beta_A}$	N_i^{β}	f_i^{β}
hyd.	VLIFYWMC	6468	0.32	1001	0.624	1721	0.499	2722	0.539
neu.	TSAGP	6783	0.343	361	0.225	914	0.265	1275	0.252
pol.	KRDNHEQ	6538	0.330	242	0.151	813	0.236	1055	0.209

Three-Class Conformational Preferences				
Class	Par.	Anti.	All beta	Par./Anti.
hydrophobic	1.91	1.53	1.65	1.25
neutral	0.66	0.77	0.74	0.85
polar	0.46	0.71	0.63	0.64

Table 5.13: Class definitions, counts, frequencies, and conformation preferences (frequency in beta divided by global frequency) for residues grouped into three classes.

I find that 977/1349 = 72% of interior residues are buried, and 945/2462 = 38% of exterior residues are buried.

5.3.4 Grouping residues into classes

When residues are grouped into three broad classes (hydrophobic, neutral, polar), the results are what I expect (hydrophobic residues prefer β_P to β_A and β to non-β; neutral and polar residues prefer β_A to β_P and non-β to β). Results are shown in Table 5.13 for all β residues and for buried beta residues. Note that hydrophobic residues prefer parallel to antiparallel. while neutral and polar residues prefer antiparallel to parallel. This is partly due to the fact that parallel structure prefers to be buried.

5.3.5 Contingency Table Analysis

The contingency table, created using the pdb_select protein set, is shown in table 5.14. Counts range from 112 (parallel exposed) to 4629 (antiparallel buried).

The loglinear models are shown in figure 5.15. G^2 and χ^2 are measures of the model error. G^2 is the likelihood ratio test statistic. The letters used to indicate marginal terms are:

1. A: Amino acid group (of 3)

Three-Way Three-Class Contingency Table				
	Antiparallel		Parallel	
	Buried	Exposed	Buried	Exposed
Hydrophobic	4629	1138	2234	156
Neutral	2093	1014	755	112
Polar	1164	1603	410	274

Table 5.14: Three-way contingency table of counts for strand direction, amino acid group, and solvent exposure.

Model	G^2	df	marginal terms						
[A]	9569	9	A						
[A][D]	5592	8	A		D				
[AD]	5441	6	A		D		AD		
[A][E][D]	2343	7	A	E	D				
[AE][D]	621	5	A	E	D	AE			
[AD][E]	2192	5	A	E	D		AD		
[AE][AD]	470	3	A	E	D	AE	AD		
[A][ED]	1786	6	A	E	D			ED	
[AE][ED]	64	4	A	E	D	AE		ED	
[AD][ED]	1635	4	A	E	D		AD	ED	
[AE][AD][ED]	21	2	A	E	D	AE	AD	ED	
[AED]	0	0	A	E	D	AE	AD	ED	AED

Table 5.15: Loglinear models, likelihood ratio test statistic (G^2), Pearson test statistic (χ^2), degrees of freedom (df), and marginal terms (A–amino acid type, of three groups, E–solvent exposure, D–direction).

2. E: Solvent accessibility (buried or exposed)

3. D: Strand direction (parallel, antiparallel)

Thus, [AE] indicates the margin which is obtained by summing over variable D, strand direction. If adding the [AE] margin to a model reduces the error, I can assume that variables A and E are not independent.

I now consider a nested hierarchy of models, where each model considered contains the previous ones as special cases. I determine the difference in G^2 between a model and its immediate predecessors. A large ΔG^2 indicates the invalidity of the assumption of independence represented by the term which distinguishes the models. With a three-dimensional contingency table, it is possible to consider all possible nested hierarchies of models; the various hierarchies differ in the order in which two-factor

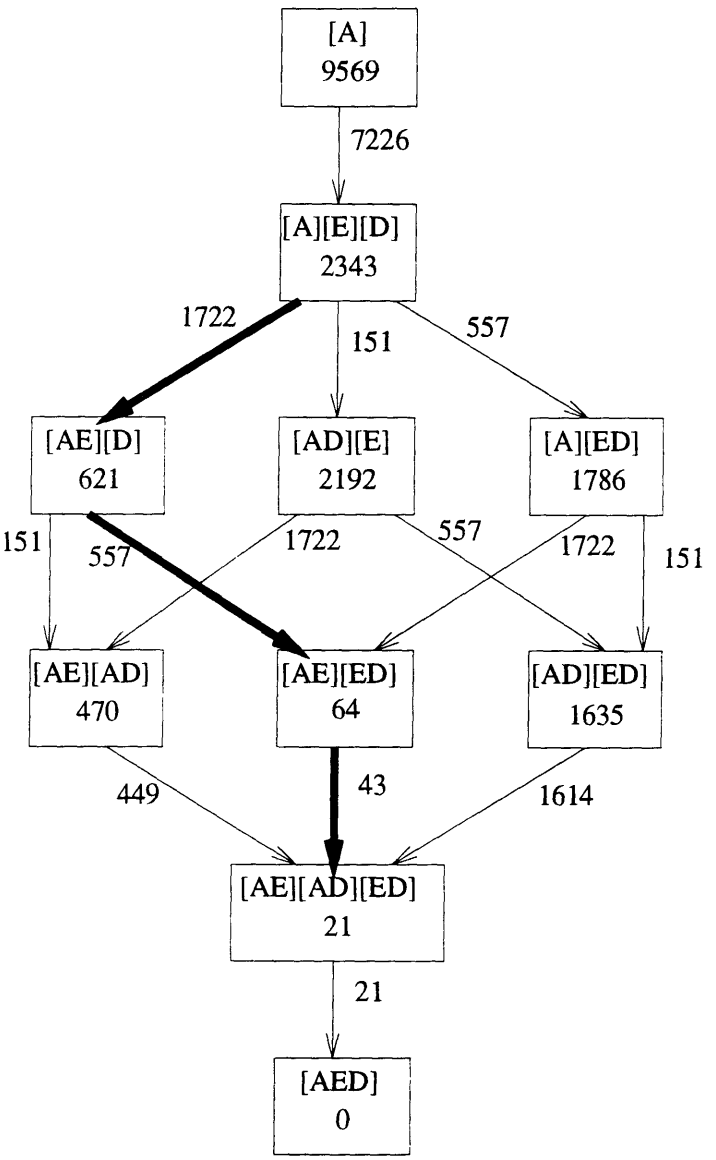


Figure 5-2: Nested model hierarchy. Each box represents a model. The model name and likelihood ratio statistic, G^2, are listed in the box. Arrows are drawn between models related by the addition of marginal terms. The arrows are annotated with the difference in G^2 and (in italics) the added marginal terms. Heavy arrows indicate the nested hierarchy which explains the most variance the earliest. Variables are A–amino acid type, E–solvent exposure, D-strand direction.

terms are added.

Figure 5-2 shows the models and the relationships between them. Consider the hierarchies between the no-two-factor (independence) model, [A][E][S], and the no-three-factor model, [AE][AD][AS]. The total difference in error is $\Delta G^2 = 2343 - 21 = 2322$. Adding margin [AE] (two degrees of freedom) to a model accounts for between 70% and 74% of the variance, depending on the choice of hierarchy. Adding margin [ED] (one degree of freedom) accounts for between 19% and 24% of the variance. And adding margin [AD] (two degrees of freedom) accounts for between 2% and 7% of the variance. Clearly, regardless of the order in which the models are examined, the [AE] margin is most effective at reducing the error, followed by the [ED] margin and then the [AD] margin.

In other words, amino acid group and solvent exposure are not independent ([AE] margin), and this effect is much greater than that for amino acid group and strand direction ([AD] margin). Moreover, we clearly see that there is a strong correlation between solvent accessibility and strand direction, and that this effect is stronger than that of association between strand direction and amino acid group!

All of the model predictions have statistically significant differences from the observed data and from their neighbors in the hierarchy. Thus all combinations of categories contain information, including the three-way interaction of amino acid group, strand direction, and solvent accessibility, above and beyond that contained in the individual pairings.

The analysis is more understandable if I compare the observed to the expected counts for each model. Table 5.16 shows the ratio of observed to expected counts for the four models in the model hierarchy {[A][E][D], [AE][D], [AE][ED], [AE][AD],[AE]}, as well as the likelihood ratio statistics and degrees of freedom. This is the sequence which reduces the variance the most quickly; the sequence is marked with heavy arrows in figure 5-2.

Model [A][E][D], which assumes independence among the three category classes, overpredicts exposed hydrophobics and buried polars, and underpredicts buried hydrophobics and exposed polars.

Model Hierarchy									
						Observed/Expected Ratios			
						Antiparallel		Parallel	
Model	G^2	df	ΔG^2	Δdf		Bur.	Exp.	Bur.	Exp.
[A][E][D]	2343	7			Hyd.	1.05	0.68	1.50	0.27
					Neu.	0.97	1.24	1.04	0.40
					Polar.	0.62	2.25	0.65	1.14
[AE][D]	621	5	1722	2	Hyd.	0.90	1.18	1.29	0.48
					Neu.	0.98	1.21	1.05	0.39
					Pol.	0.99	1.14	1.03	0.58
[AE][ED]	64	4	557	1	Hyd.	0.97	1.01	1.08	0.96
					Neu.	1.05	1.03	0.88	0.79
					Pol.	1.06	0.98	0.86	1.16
[AE][AD][ED]	21	2	43	2	Hyd.	0.99	1.03	1.01	0.84
					Neu.	0.99	1.01	1.02	0.90
					Pol.	1.04	0.97	0.91	1.18

Table 5.16: A hierarchy of models presented in decreasing order of ΔG^2. "df" is degrees of freedom. Also shown are the ratios of observed counts to expected counts (as predicted by the models).

Model [AE][D] removes the independence assumption for amino acid class and solvent accessibility, by including the [AE] pairwise term. Clearly, the tendency for a given exposure category to overpredict for one amino acid class and underpredict for another has disappeared. However, this model now overpredicts parallel exposed, and underpredicts antiparallel exposed. This is because we are still assuming that the model should be the same for parallel and antiparallel beta pairs.

Including terms to represent the dependency between the amino acid class and the solvent exposure yields model [AE][ED]. This allows us to account for the fact that parallel sheets prefer to be buried. This model predicts all the antiparallel counts well, but doesn't do well on parallel exposed (neutral overpredicted and polar underpredicted), and overpredicts parallel buried (neutral and polar). [AE][AD][ED], with all three pairwise terms, also does fine on the antiparallel prediction, but still doesn't have the parallel counts quite right.

What happens when we don't consider solvent exposure at all? Notice that model [AD] is significantly better than model [A][D] ($\Delta G^2 = 151$, with 2 degrees of freedom difference; $P = 0.0$). With this pair of models, I am testing whether amino acid group

is independent of strand direction, and I find that it is not. Thus, separating parallel from antiparallel structure in the secondary structure representation may be justified, particularly in the case where solvent exposure is not considered or predicted. On the other hand, in an application like threading, the parallel/antiparallel distinction has relatively little information compared to the buried/exposed distinction.

Another point is that the contingency table groups amino acids into three broad classes. Much of the variance is explained by the amino acid group without having to consider the individual amino acid type.

5.4 Implications

5.4.1 Protein folding and structure

My results show that the difference in amino acid composition between parallel and antiparallel sheets is partly due to the fact that parallel sheets are more buried. In addition, I find that there is a stronger requirement in parallel sheets for beta-branched residues, and a stronger requirement in antiparallel sheets for large hydrophobic residues (Phe and Tyr). Moreover, I find that the segregation of hydrophobic residues to the inside of the protein is more important in folding than choosing whether a strand is parallel or antiparallel.

5.4.2 Secondary structure prediction

Finding a way to incorporate solvent exposure may improve the accuracy of beta sheet prediction. Predicting solvent exposure may be a productive first step in secondary structure prediction. This approach has not been explored much to date.

Where secondary structure predictions are to be used in further tertiary structure predictions, it may make sense to predict solvent exposure *instead of*, or along with, predicting alpha/beta/coil classifications. The inside/outside distinction is a natural classification for several reasons. Some researchers claim that atomic solvation can distinguish correctly folded from misfolded proteins [Baumann *et al.*, 1989, Chiche

et al., 1990. Holm and Sander, 1992, Vila *et al.*, 1991]. Some of the inverse folding methods consider solvent exposure as part of the structure description [Jones *et al.*, 1992, White *et al.*, 1994, Lathrop *et al.*, , Luthy *et al.*, 1992, Johnson *et al.*, 1993]. There exist prediction algorithms for the interior/exterior classification [Benner *et al.*, 1994].

5.4.3 Tertiary structure prediction

Given the protein sequence and the locations of beta strands, knowing or predicting solvent exposure should help predict whether strand connections are parallel or antiparallel.

There is generally a tradeoff, due to the limited sample size of known protein strutures. between how fine the structure representation categories can be, and the accuracy of predictions based on empirical observations. My results suggest that in threading methods for tertiary structure prediction, if one is presented with a choice between representing strand direction or solvent exposure, the latter should be used.

Chapter 6

Pairwise Interactions in Beta Sheets

6.1 Introduction

In this chapter I examine residue pairs that occur in specific topological relationships to each other. This is a specialization of the residue pairs representation. This specialization of the residue pair representation might prove useful in determining protein structure. The topological relationships are defined by the hydrogen bonding patterns in beta sheets.

Lifson and Sander in their 1980 statistical analysis of interactions between side chains in beta sheets, found that there was significant "recognition" between side chains [Lifson and Sander, 1980]. In this chapter, I repeat and extend their analysis. In looking at the occurrence of amino acids pairs in beta sheets, I find the following:

- The observed counts make intuitive sense. Hydrophobes pair with hydrophobes; neutrals and polars with themselves and each other; and opposite charges attract.
- There is specific pairwise recognition of beta, $(i, i + 2)$, and diagonal pairs in beta sheets.
- There is nonspecific pairwise recognition of beta, $(i, i + 2)$, and diagonal pairs

Class	Residues
Hydrophobic	VLIFYWMC
Neutral	TSAGP
Polar	KRDNHEQ

Table 6.1: Amino acid classification into three hydrophobicity classes.

in beta sheets.

- Some of the Lifson/Sander pairwise recognition can be explained by solvent exposure (polar atoms congregate to surface, and therefore appear as pairs more often). Considering solvent exposure reduces the observed association of residues significantly.

- Considering strand direction (parallel or antiparallel) has much less of an effect on amino acid pairing than does solvent exposure.

- $(i, j+1)$ pairings in buried beta structure show unexpected nonrandomness.

6.2 Method

The counts are defined in Section 5.2.3. The frequency of residue type i in a given topological relationship (say, β for beta pairs) is $f_i^\beta = N_i^\beta/N$. The frequency of pairs of type i and j is $f_{ij} = 2N_{ij}/N$. The likelihood ratio between the observed and expected pair frequencies is

$$R_{ij} = \frac{2N_{ij}N}{N_i N_j}.$$

I counted beta pairs (as defined by the DSSP program) in the 252 proteins listed in Section B.2.2. The amino acid groupings are those used by Lifson and Sander in analyzing nonspecific beta pair data, as shown in Table 6.1.

A residue is considered to be buried if its DSSP accessibility is less than 20% of its maximum accessibility, and it is exposed if its DSSP accessibility is greater than 20% of its maximum accessibility. The maximum accessibilities are listed in section B.3.

For each beta pair, I determine the amino acid group and exposure of each residue, and whether the strands are parallel or antiparallel.

I consider a five-dimensional contingency table, shown in Table 6.15. The dimensions are the amino acid tyep and solvent exposure of each residue, and the strand direction.

6.2.1 Significance testing

χ^2 analysis to test for a significant difference between observed and expected pairwise frequency distributions is performed as described by Lifson and Sander [Lifson and Sander, 1980]. The expected number of (i, j) pairs, based on the hypothesis of random pairing, is $E_{ij} = f_i f_j N/2$. The observed number of (i, j) pairs is

6.3 Results and Discussion

6.3.1 Counts and preferences

Table 6.2 contains beta-pair counts and frequency ratios for parallel, antiparallel, and both together, for amino acids grouped into the classes hydrophobic, neutral and polar. Cells were tested individually for significant differences between observed and expected (by random pairing) counts, and each of the three tables was tested as a whole table for difference from an expected table generated based on random pairing. The frequency ratios of observed to expected counts indicate whether a pairing is favored (> 1.0) or disfavored (< 1.0). The tendencies for pairing preferences are the same for both parallel and antiparallel, though the observed/expected ratios are more extreme for the parallel beta pairs. In general, the following pairs are favored: HH, NN, PP, NP, where H is hydrophobic, N is neutral, and P is polar. On the other hand, HP is disfavored. NP shows no significant difference from random association. This is basically what we would expect in terms of pairings.

Tables 6.3, 6.4, and 6.5 contain the beta pair counts and frequency ratios for parallel, antiparallel and all-beta, for the amino acids separated into all 20 types.

			H	N	P
Parallel	Observed Counts	H	686	213	104
		N	213	122	66
		P	104	66	84
	Observed/Expected	H	1.13	0.88	**0.68**
		N		1.26	1.07
		P			**2.16**
Antiparallel	Observed counts	H	1312	516	440
		N	516	346	283
		P	440	283	346
	Observed/Expected	H	**1.14**	0.89	**0.81**
		N		1.18	1.04
		P			**1.36**
Both	Observed Counts	H	2008	732	548
		N	732	468	351
		P	548	351	432
	Observed/Expected	H	**1.15**	0.89	**0.77**
		N		**1.20**	1.05
		P			**1.50**

Table 6.2: Beta pair counts and preferences for parallel, antiparallel, and all-beta, for amino acids grouped into three classes. Protein set: Rost and Sander. Cells that are significantly different than expected by random association are underlined (P< .05) and in bold face (P< .001). Significances of tables: Parallel, $\chi^2 = 89.1$; Antiparallel, $\chi^2 = 91.9$; Both, $\chi^2 = 174.4$. Each of these is significant at P= 0 for 3 degrees of freedom.

Parallel Beta Pair Counts

	G	P	D	E	A	N	Q	S	T	K	R	H	V	I	M	C	L	F	Y	W
G	2	1	2	5	10	2	2	5	8	2	1	0	14	11	2	2	12	8	5	3
P	1	0	0	0	0	0	0	2	2	0	0	0	3	2	3	0	2	0	1	0
D	2	0	0	1	3	3	0	5	4	1	0	3	3	3	1	1	4	2	0	0
E	5	0	1	0	1	1	1	3	1	9	5	1	3	5	1	1	1	3	3	1
A	10	0	3	1	8	2	2	6	7	3	0	1	22	21	3	3	17	7	4	3
N	2	0	3	1	2	2	2	1	6	3	1	1	6	1	2	0	1	2	3	0
Q	2	0	0	1	2	2	2	1	1	0	1	1	6	2	0	0	2	1	4	1
S	5	2	5	3	6	1	1	6	8	5	0	1	12	2	2	1	5	2	2	1
T	8	2	4	1	7	6	1	8	8	3	5	4	10	13	4	2	7	1	1	0
K	2	0	1	9	3	3	0	5	3	0	2	2	2	5	0	0	4	3	1	1
R	1	0	0	5	0	1	1	0	5	2	0	1	2	3	0	1	5	2	2	2
H	0	0	3	1	1	1	1	1	4	2	1	2	1	3	0	0	3	0	0	1
V	14	3	3	3	22	6	6	12	10	2	2	1	86	52	7	4	43	19	7	4
I	11	2	3	5	21	1	2	2	13	5	3	3	52	58	5	3	39	11	13	6
M	2	3	1	1	3	2	0	2	4	0	0	0	7	5	0	0	5	0	2	1
C	2	0	1	1	3	0	0	1	2	0	1	0	4	3	0	0	2	3	0	1
L	12	2	4	1	17	1	2	5	7	4	5	3	43	39	5	2	32	14	6	2
F	8	0	2	3	7	2	1	2	1	3	2	0	19	11	0	3	14	4	1	2
Y	5	1	0	3	4	3	4	2	1	1	2	0	7	13	2	0	6	1	2	0
W	3	0	0	1	3	0	1	1	0	1	2	1	4	6	1	1	2	2	0	0

Parallel Beta Pair Likelihood Ratios

	G	P	D	E	A	N	Q	S	T	K	R	H	V	I	M	C	L	F	Y	W
G	0.4	1.1	0.9	1.9	1.4	0.9	1.2	1.2	1.4	0.7	0.5	0.0	0.8	0.7	0.9	1.4	1.0	1.6	1.5	1.8
P		0.0	0.0	0.0	0.0	0.0	0.0	3.0	2.2	0.0	0.0	0.0	1.0	0.8	**8.2**	0.0	1.0	0.0	1.8	0.0
D			0.0	1.0	1.1	3.5	0.0	3.3	1.9	1.0	0.0	**5.5**	0.5	0.5	1.2	1.9	0.9	1.1	0.0	0.0
E				0.0	0.3	0.9	1.2	1.5	0.4	**7.1**	**5.5**	1.4	0.4	0.7	0.9	1.5	0.2	1.3	1.9	1.2
A					0.9	0.7	0.9	1.2	1.0	0.9	0.0	0.5	1.0	1.1	1.1	1.7	1.1	1.1	0.9	1.4
N						2.2	2.9	0.6	2.7	2.8	1.3	1.7	0.8	0.2	2.2	0.0	0.2	1.0	2.2	0.0
Q							3.9	0.8	0.6	0.0	1.7	2.3	1.1	0.4	0.0	0.0	0.6	0.7	4.0	2.0
S								2.0	2.0	2.6	0.0	0.9	0.9	0.2	1.2	1.0	0.6	0.6	0.8	0.8
T									1.5	1.1	2.6	2.8	0.6	0.9	1.8	1.5	0.6	0.2	0.3	0.0
K										0.0	2.2	2.9	0.2	0.7	0.0	0.0	0.7	1.3	0.6	1.2
R											0.0	2.0	0.3	0.6	0.0	2.1	1.2	1.2	1.8	3.5
H												5.3	0.2	0.8	0.0	0.0	1.0	0.0	0.0	2.3
V													**1.5**	1.1	1.0	0.9	1.1	1.2	0.7	0.7
I														1.4	0.8	0.8	1.2	0.8	1.5	1.3
M															0.0	0.0	1.1	0.0	1.5	1.5
C																0.0	0.7	2.4	0.0	2.4
L																	1.3	1.3	0.8	0.6
F																		0.9	0.3	1.3
Y																			1.0	0.0
W																				0.0

Table 6.3: Counts $N_{ij}^{\beta_P}$ and frequencies $f_{ij}^{\beta_P}$ of parallel beta pairs. Protein set: Rost and Sander. $\chi^2 = 339.0$ with 190 degrees of freedom; $P < 10^{-9}$. Cells that are significant are underlined (P< .05) or in bold face (P< .001).

Antiparallel Beta Pair Counts

	G	P	D	E	A	N	Q	S	T	K	R	H	V	I	M	C	L	F	Y	W
G	10	2	11	4	13	6	1	19	13	7	5	9	33	17	4	6	14	13	17	9
P	2	0	1	3	1	3	3	3	4	3	4	1	9	4	5	4	8	4	7	1
D	11	1	4	3	6	5	5	9	12	11	7	5	9	8	4	3	9	5	5	3
E	4	3	3	6	6	3	3	9	12	35	19	9	15	9	3	2	11	6	4	0
A	13	1	6	6	18	4	6	10	24	12	3	6	33	26	7	7	23	23	12	8
N	6	3	5	3	4	6	3	13	14	7	1	4	8	4	3	6	6	8	11	2
Q	1	3	5	3	6	3	12	9	16	11	7	1	10	12	1	3	11	12	11	2
S	19	3	9	9	10	13	9	28	34	15	7	7	30	14	7	4	18	11	17	3
T	13	4	12	12	24	14	16	34	44	30	18	8	33	21	7	6	17	13	18	3
K	7	3	11	35	12	7	11	15	30	8	3	2	25	23	8	9	15	9	18	7
R	5	4	7	19	3	1	7	7	18	3	2	7	22	14	4	2	10	9	8	5
H	9	1	5	9	6	4	1	7	8	2	7	6	10	9	1	4	7	4	10	1
V	33	9	9	15	33	8	10	30	33	25	22	10	80	65	9	10	66	45	31	7
I	17	4	8	9	26	4	12	14	21	23	14	9	65	22	5	15	50	21	21	10
M	4	5	4	3	7	3	1	7	7	8	4	1	9	5	4	3	16	8	9	7
C	6	4	3	2	7	6	3	4	6	9	2	4	10	15	3	26	8	14	8	5
L	14	8	9	11	23	6	11	18	17	15	10	7	66	50	16	8	62	32	22	16
F	13	4	5	6	23	8	12	11	13	9	9	4	45	21	8	14	32	28	18	3
Y	17	7	5	4	12	11	11	17	18	18	8	10	31	21	9	8	22	18	18	10
W	9	1	3	0	8	2	2	3	3	7	5	1	7	10	7	5	16	3	10	4

Antiparallel Beta Pair Likelihood Ratios

	G	P	D	E	A	N	Q	S	T	K	R	H	V	I	M	C	L	F	Y	W
G	1.0	0.6	1.9	0.5	1.1	1.1	0.2	1.5	0.8	0.6	0.7	1.7	1.3	1.0	0.7	0.9	0.7	1.0	1.3	1.8
P		0.0	0.5	1.2	0.3	1.6	1.4	0.7	0.7	0.7	1.6	0.6	1.0	0.7	2.8	1.8	1.2	0.9	1.6	0.6
D			1.1	0.7	0.9	1.5	1.3	1.2	1.2	1.5	1.6	1.6	0.6	0.8	1.2	0.7	0.8	0.6	0.7	1.0
E				1.0	0.7	0.7	0.6	0.9	1.0	**3.8**	**3.3**	2.2	0.8	0.7	0.7	0.4	0.7	0.6	0.4	0.0
A					1.3	0.6	0.8	0.7	1.2	0.8	0.3	1.0	1.1	1.3	1.1	0.9	1.0	1.5	0.8	1.4
N						2.0	0.8	1.9	1.5	1.0	0.2	1.4	0.6	0.4	1.0	1.6	0.5	1.1	1.5	0.7
Q							**2.8**	1.1	1.5	1.4	1.4	0.3	0.6	1.0	0.3	0.7	0.8	1.4	1.3	0.6
S								1.8	1.6	1.0	0.7	1.1	0.9	0.6	1.0	0.5	0.7	0.6	1.0	0.5
T									**1.6**	1.5	1.5	0.9	0.8	0.7	0.8	0.5	0.5	0.6	0.8	0.4
K										0.5	0.3	0.3	0.8	1.1	1.2	1.1	0.6	0.5	1.1	1.1
R											0.4	1.8	1.1	1.1	1.0	0.4	0.7	0.9	0.8	1.3
H												2.2	0.7	1.0	0.4	1.1	0.7	0.6	1.5	0.4
V													1.2	1.4	0.6	0.6	1.3	1.3	0.9	0.5
I														0.7	0.5	1.3	1.4	0.9	0.9	1.1
M															1.4	0.8	1.5	1.1	1.3	2.6
C																**5.5**	0.6	1.5	0.9	1.5
L																	**1.6**	1.2	0.9	1.6
F																		1.5	1.0	0.4
Y																			1.1	1.5
W																				1.6

Table 6.4: Counts and frequencies of antiparallel beta pairs. Protein set: Rost and Sander. $\chi^2 = 545.9$ with 190 degrees of freedom. Cells that are significant are underlined (P< .05) and in bold face (P< .001).

All Beta Pair Counts

	G	P	D	E	A	N	Q	S	T	K	R	H	V	I	M	C	L	F	Y	W
G	12	3	13	9	23	8	3	24	21	9	6	9	47	28	6	9	26	21	23	12
P	3	0	1	3	1	3	3	5	6	3	4	1	12	6	8	4	10	4	8	1
D	13	1	4	4	9	8	5	14	16	12	7	8	12	11	5	4	13	7	5	3
E	9	3	4	6	7	4	4	12	13	44	24	10	18	14	4	3	12	9	9	1
A	23	1	9	7	26	6	9	16	31	15	3	7	55	47	10	10	40	30	16	11
N	8	3	8	4	6	8	5	14	20	10	2	5	14	5	5	6	7	10	15	2
Q	3	3	5	4	9	5	16	10	17	11	8	2	16	14	1	3	13	13	15	3
S	24	5	14	12	16	14	10	34	42	20	8	8	42	16	9	5	23	14	19	4
T	21	6	16	13	31	20	17	42	52	33	23	12	43	34	11	8	24	14	19	3
K	9	3	12	44	15	10	11	20	33	8	5	4	28	28	8	9	19	12	19	8
R	6	4	7	24	3	2	8	8	23	5	2	8	24	17	4	3	15	11	10	7
H	9	1	8	10	7	5	2	8	12	4	8	8	11	12	1	4	10	4	10	2
V	47	12	12	18	55	14	16	42	43	28	24	11	166	118	17	14	109	64	38	11
I	28	6	11	14	47	5	14	16	34	28	17	12	118	82	10	18	89	32	34	16
M	6	8	5	4	10	5	1	9	11	8	4	1	17	10	4	3	21	8	11	8
C	9	4	4	3	10	6	3	5	8	9	3	4	14	18	3	26	10	17	8	6
L	26	10	13	12	40	7	13	23	24	19	15	10	109	89	21	10	94	47	28	18
F	21	4	7	9	30	10	13	14	14	12	11	4	64	32	8	17	47	32	20	5
Y	23	8	5	9	16	15	15	19	19	19	10	10	38	34	11	8	28	20	20	10
W	12	1	3	1	11	2	3	4	3	8	7	2	11	16	8	6	18	5	10	4

All Beta Pair Likelihood Ratios

	G	P	D	E	A	N	Q	S	T	K	R	H	V	I	M	C	L	F	Y	W
G	0.8	0.7	1.6	0.8	1.2	1.0	0.3	1.4	0.9	0.6	0.6	1.3	1.1	0.9	0.8	1.0	0.8	1.1	1.3	1.8
P		0.0	0.4	1.0	0.2	1.4	1.3	1.1	1.0	0.7	1.5	0.5	1.0	0.7	**3.7**	1.7	1.1	0.8	1.7	0.5
D			1.0	0.7	0.9	2.0	1.1	1.6	1.4	1.5	1.4	2.3	0.5	0.7	1.2	0.9	0.8	0.7	0.6	0.9
E				0.8	0.6	0.7	0.7	1.0	0.9	**4.2**	**3.7**	2.2	0.6	0.7	0.8	0.5	0.6	0.7	0.8	0.2
A					1.2	0.6	0.9	0.8	1.2	0.8	0.3	0.9	1.1	1.2	1.1	1.0	1.1	1.3	0.8	1.4
N						2.0	1.1	1.6	1.8	1.3	0.4	1.4	0.6	0.3	1.3	1.4	0.4	1.1	1.7	0.6
Q							**3.4**	1.1	1.4	1.3	1.5	0.5	0.7	0.8	0.2	0.6	0.7	1.3	1.6	0.8
S								**1.8**	**1.7**	1.2	0.8	1.1	0.9	0.5	1.1	0.5	0.7	0.7	1.0	0.5
T									**1.6**	1.5	1.7	1.2	0.7	0.8	1.0	0.7	0.5	0.5	0.8	0.3
K										0.5	0.5	0.6	0.7	0.9	1.1	1.1	0.6	0.6	1.1	1.2
R											0.3	1.9	0.9	0.9	0.8	0.6	0.8	1.0	1.0	1.7
H												2.7	0.6	0.9	0.3	1.1	0.7	0.5	1.3	0.7
V													**1.4**	1.3	0.8	0.6	1.2	1.2	0.8	0.6
I														1.3	0.6	1.0	1.4	0.8	1.0	1.2
M															1.0	0.7	1.3	0.9	1.3	2.4
C																**5.6**	0.6	1.6	0.9	1.6
L																	**1.5**	1.2	0.8	1.3
F																		1.4	1.0	0.6
Y																			1.1	1.4
W																				1.4

Table 6.5: Counts and frequencies of all beta pairs. Protein set: Rost and Sander. Overall chi^2 = 721.5 with 190 degrees of freedom. Underlined likelihood ratios are significant at P< .05; bold face P< .001.

Table 6.6 contains counts and frequency ratios for $(i, i+2)$ beta pairs. Table 6.7 is the same, but for grouped amino acids.

Tables 6.8, 6.9, and 6.10 contain counts and frequency ratios for diagonal pairs. Table 6.11 is the same, but for grouped amino acids.

$i, i+2$ Counts																				
	G	P	D	E	A	N	Q	S	T	K	R	H	V	I	M	C	L	F	Y	W
G	20	9	5	11	30	10	12	13	25	11	12	8	41	39	13	8	38	22	23	9
P	9	0	0	1	8	2	2	5	11	6	7	2	17	10	5	3	9	10	5	4
D	5	0	6	7	6	3	8	9	13	7	9	7	30	13	2	4	10	7	5	7
E	11	1	7	12	16	3	6	15	15	10	5	11	29	25	3	5	18	15	4	7
A	30	8	6	16	30	8	5	17	18	20	11	11	52	39	10	8	40	20	21	7
N	10	2	3	3	8	4	4	9	14	7	2	6	28	14	3	5	14	12	6	4
Q	12	2	8	6	5	4	2	9	24	6	5	5	17	12	4	3	18	10	11	1
S	13	5	9	15	17	9	9	44	48	13	12	9	40	25	7	9	30	25	19	7
T	25	11	13	15	18	14	24	48	66	24	9	8	61	29	8	7	35	15	28	3
K	11	6	7	10	20	7	6	13	24	16	11	6	40	27	10	7	33	15	23	5
R	12	7	9	5	11	2	5	12	9	11	10	3	17	24	4	7	20	10	18	5
H	8	2	7	11	11	6	5	9	8	6	3	8	17	8	2	2	14	7	4	2
V	41	17	30	29	52	28	17	40	61	40	17	17	120	84	12	16	84	56	30	14
I	39	10	13	25	39	14	12	25	29	27	24	8	84	48	17	13	59	33	20	10
M	13	5	2	3	10	3	4	7	8	10	4	2	12	17	8	6	14	14	8	1
C	8	3	4	5	8	5	3	9	7	7	7	2	16	13	6	0	24	4	15	1
L	38	9	10	18	40	14	18	30	35	33	20	14	84	59	14	24	82	36	30	18
F	22	10	7	15	20	12	10	25	15	15	10	7	56	33	14	4	36	32	18	8
Y	23	5	5	4	21	6	11	19	28	23	18	4	30	20	8	15	30	18	8	3
W	9	4	7	7	7	4	1	7	3	5	5	2	14	10	1	1	18	8	3	2

$i, i+2$ Likelihood Ratios																				
	G	P	D	E	A	N	Q	S	T	K	R	H	V	I	M	C	L	F	Y	W
G	0.9	1.3	0.5	0.9	1.3	1.1	1.2	0.6	0.9	0.6	1.0	1.0	0.9	1.2	1.5	0.9	1.0	1.0	1.3	1.3
P		0.0	0.0	0.2	1.1	0.7	0.6	0.7	1.3	1.1	1.8	0.7	1.1	1.0	1.7	1.1	0.8	1.4	0.9	1.8
D			1.5	1.2	0.6	0.7	1.9	0.9	1.1	0.9	1.7	1.9	1.4	0.9	0.5	1.0	0.6	0.7	0.6	2.3
E				1.5	1.2	0.5	1.0	1.1	0.9	0.9	0.7	2.2	1.0	1.3	0.6	0.9	0.8	1.1	0.4	1.7
A					1.3	0.8	0.5	0.8	0.6	1.1	0.9	1.3	1.0	1.1	1.1	0.9	1.0	0.9	1.1	1.0
N						1.0	0.9	0.9	1.2	0.9	0.4	1.6	1.3	1.0	0.8	1.3	0.9	1.3	0.8	1.3
Q							0.5	0.9	1.9	0.7	0.9	1.3	0.8	0.8	1.0	0.8	1.1	1.0	1.4	0.3
S								**2.0**	**1.7**	0.7	1.0	1.1	0.8	0.8	0.8	1.0	0.8	1.1	1.1	1.0
T									**1.9**	1.1	0.6	0.8	1.0	0.7	0.7	0.6	0.7	0.5	1.2	0.3
K										1.1	1.1	0.9	1.0	1.0	1.4	1.0	1.1	0.8	1.6	0.9
R											1.5	0.6	0.6	1.3	0.8	1.4	1.0	0.8	1.8	1.3
H												2.5	0.9	0.6	0.6	0.6	1.0	0.8	0.6	0.7
V													1.1	1.2	0.6	0.8	1.0	1.1	0.8	0.9
I														1.0	1.2	1.0	1.0	1.0	0.7	0.9
M															2.1	1.6	0.9	1.5	1.1	0.3
C																0.0	1.6	0.4	2.1	0.4
L																	1.3	0.9	1.0	1.5
F																		1.4	1.0	1.1
Y																			0.5	0.5
W																				0.9

Table 6.6: Counts and frequencies of $(i, i+2)$ pairs

Observed Counts	H	1596	801	667
	N	801	528	349
	P	667	349	320
Observed/Expected	H	1.03	0.95	0.99
	N		<u>1.07</u>	1.00
	P			1.01

Table 6.7: $(i, i+2)$ counts for residues grouped by hydrophobicity. Protein set: Rost and Sander. Underlined cell is significant; P= 0.0026. Total $\chi^2 = 16.63$; this is significant, P$< 10^{-4}$.

Parallel Diagonal Counts

	G	P	D	E	A	N	Q	S	T	K	R	H	V	I	M	C	L	F	Y	W
G	4	1	2	5	7	2	2	3	3	1	2	2	11	14	0	3	12	3	0	1
P		0	0	0	0	0	0	1	1	0	0	0	4	5	0	0	0	0	0	0
D			0	0	3	1	0	2	0	4	0	2	8	3	0	0	1	2	0	1
E				0	4	0	0	2	7	2	2	2	6	4	1	0	1	0	2	2
A					2	3	1	6	2	2	3	3	25	20	5	1	12	8	6	1
N						1	0	1	3	1	2	0	7	5	2	0	0	2	2	0
Q							0	0	1	0	0	1	4	1	0	0	4	1	1	0
S								1	5	4	2	0	5	5	2	0	10	3	4	1
T									4	4	1	2	16	9	1	1	15	3	1	1
K										1	0	0	11	7	1	0	8	0	2	0
R											1	0	9	2	1	0	3	1	4	1
H												0	2	1	1	0	1	2	3	0
V													35	49	9	7	42	14	8	2
I														20	7	4	34	10	8	3
M															1	0	2	3	0	0
C																0	2	1	0	0
L																	11	10	5	3
F																		2	5	1
Y																			1	1
W																				0

Parallel Diagonal Likelihood Ratios

	G	P	D	E	A	N	Q	S	T	K	R	H	V	I	M	C	L	F	Y	W
G	1.8	1.5	1.3	2.3	1.1	1.1	2.3	0.9	0.7	0.4	1.0	1.7	0.7	1.1	0.0	2.9	1.2	0.8	0.0	1.0
P		0.0	0.0	0.0	0.0	0.0	0.0	2.2	1.5	0.0	0.0	0.0	1.6	2.7	0.0	0.0	0.0	0.0	0.0	0.0
D			0.0	0.0	1.3	1.6	0.0	1.8	0.0	4.2	0.0	4.7	1.3	0.7	0.0	0.0	0.3	1.4	0.0	2.9
E				0.0	1.3	0.0	0.0	1.3	3.1	1.5	2.1	3.4	0.7	0.7	1.0	0.0	0.2	0.0	1.4	4.2
A					0.4	1.2	0.8	1.3	0.3	0.5	1.1	1.8	1.0	1.1	1.8	0.7	0.8	1.4	1.4	0.7
N						2.8	0.0	0.8	1.6	0.9	2.6	0.0	1.0	1.0	2.5	0.0	0.0	1.2	1.7	0.0
Q							0.0	0.0	1.1	0.0	0.0	4.3	1.2	0.4	0.0	0.0	2.0	1.3	1.7	0.0
S								0.9	1.5	2.1	1.5	0.0	0.4	0.6	1.4	0.0	1.4	1.1	1.9	1.4
T									1.7	1.5	0.5	1.6	0.9	0.7	0.5	0.9	1.4	0.7	0.3	1.0
K										1.3	0.0	0.0	1.1	0.9	0.8	0.0	1.3	0.0	1.1	0.0
R											2.5	0.0	1.3	0.4	1.2	0.0	0.7	0.6	3.2	2.4
H												0.0	0.4	0.3	1.8	0.0	0.4	1.9	3.8	0.0
V													1.1	1.0	1.2	1.8	1.1	0.9	0.7	0.5
I														1.1	1.2	1.4	1.2	0.9	1.0	1.1
M															2.2	0.0	0.4	1.7	0.0	0.0
C																0.0	0.8	1.1	0.0	0.0
L																	0.9	1.1	0.7	1.3
F																		1.1	1.9	1.1
Y																			1.0	1.5
W																				0.0

Table 6.8: Parallel diagonal pairs

Antiparallel Diagonal Counts

	G	P	D	E	A	N	Q	S	T	K	R	H	V	I	M	C	L	F	Y	W
G	2	2	2	5	14	5	11	12	11	13	14	3	20	9	4	4	14	15	15	3
P		1	2	2	2	1	4	2	4	5	0	0	6	3	1	1	3	3	0	0
D			1	5	6	6	1	5	10	4	4	2	14	5	2	1	4	2	6	2
E				1	5	4	0	6	9	13	9	5	11	8	2	3	9	8	9	1
A					10	4	6	22	12	5	5	7	35	23	9	3	17	17	18	4
N						0	3	9	10	5	6	1	8	3	0	2	6	2	5	1
Q							2	12	13	4	4	3	8	6	1	3	9	4	6	4
S								22	22	13	13	7	24	10	7	3	19	9	4	3
T									24	10	8	9	36	19	5	8	15	8	9	2
K										3	6	4	19	14	2	4	14	8	11	5
R											1	6	16	7	6	6	10	4	11	2
H												1	16	6	0	3	6	5	5	0
V													34	42	14	20	52	31	27	11
I														18	4	6	26	27	14	5
M															3	3	5	5	6	0
C																2	16	5	6	1
L																	26	34	20	10
F																		10	16	8
Y																			9	13
W																				3

Antiparallel Diagonal Likelihood Ratios

	G	P	D	E	A	N	Q	S	T	K	R	H	V	I	M	C	L	F	Y	W
G	0.4	0.9	0.5	0.9	1.2	1.2	2.1	1.0	0.8	1.6	2.0	0.7	0.8	0.7	1.0	0.8	0.8	1.3	1.4	0.7
P		3.9	2.0	1.4	0.7	1.0	3.1	0.7	1.2	2.5	0.0	0.0	1.0	0.9	1.0	0.8	0.7	1.1	0.0	0.0
D			1.0	1.8	1.1	3.1	0.4	0.9	1.6	1.0	1.2	0.9	1.2	0.8	1.0	0.4	0.5	0.4	1.1	1.0
E				0.5	0.7	1.5	0.0	0.7	1.0	2.4	2.0	1.7	0.7	0.9	0.7	0.9	0.8	1.1	1.3	0.4
A					1.3	0.8	0.9	1.4	0.7	0.5	0.5	1.2	1.1	1.3	1.7	0.4	0.8	1.1	1.3	0.8
N						0.0	1.2	1.6	1.6	1.3	1.9	0.5	0.7	0.5	0.0	0.9	0.8	0.4	1.0	0.5
Q							1.3	1.6	1.6	0.8	1.0	1.1	0.6	0.7	0.4	1.0	0.9	0.6	0.9	1.7
S								2.6	1.2	1.1	1.4	1.1	0.7	0.5	1.2	0.4	0.8	0.6	0.3	0.5
T									2.4	0.8	0.8	1.3	1.0	0.9	0.8	1.0	0.6	0.5	0.5	0.3
K										0.8	0.9	1.0	0.9	1.1	0.5	0.8	0.9	0.7	1.1	1.3
R											0.4	1.7	0.9	0.7	1.9	1.5	0.8	0.4	1.3	0.6
H												0.9	1.3	0.9	0.0	1.2	0.7	0.9	0.9	0.0
V													1.1	1.1	1.3	1.5	1.1	1.0	0.9	1.0
I														1.7	0.6	0.8	1.0	1.5	0.8	0.8
M															3.2	1.3	0.6	0.9	1.2	0.0
C																1.4	1.6	0.8	1.0	0.4
L																	1.6	1.5	1.0	1.3
F																		1.3	1.1	1.5
Y																			1.3	2.6
W																				3.3

Table 6.9: Antiparallel diagonal pairs

Parallel Diagonal Counts

	G	P	D	E	A	N	Q	S	T	K	R	H	V	I	M	C	L	F	Y	W
G	6	3	4	10	21	7	13	15	14	14	16	5	31	23	4	7	26	18	15	4
P		1	2	2	2	1	4	3	5	5	0	0	10	8	1	1	3	3	0	0
D			1	5	9	7	1	7	10	8	4	4	22	8	2	1	5	4	6	3
E				1	9	4	0	8	16	15	11	7	17	12	3	3	10	8	11	3
A					12	7	7	28	14	7	8	10	60	43	14	4	29	25	24	5
N						1	3	10	13	6	8	1	15	8	2	2	6	4	7	1
Q							2	12	14	4	4	4	12	7	1	3	13	5	7	4
S								23	27	17	15	7	29	15	9	3	29	12	8	4
T									28	14	9	11	52	28	6	9	30	11	10	3
K										4	6	4	30	21	3	4	22	8	13	5
R											2	6	25	9	7	6	13	5	15	3
H												1	18	7	1	3	7	7	8	0
V													69	91	23	27	94	45	35	13
I														38	11	10	60	37	22	8
M															4	3	7	8	6	0
C																2	18	6	6	1
L																	37	44	25	13
F																		12	21	9
Y																			10	14
W																				3

Parallel Diagonal Likelihood Ratios

	G	P	D	E	A	N	Q	S	T	K	R	H	V	I	M	C	L	F	Y	W
G	0.9	1.1	0.7	1.2	1.2	1.2	2.1	1.0	0.8	1.3	1.8	0.9	0.8	0.9	0.7	1.1	1.0	1.1	1.1	0.8
P		3.4	1.6	1.2	0.5	0.8	3.0	0.9	1.3	2.2	0.0	0.0	1.2	1.5	0.8	0.8	0.5	0.9	0.0	0.0
D			0.8	1.4	1.1	2.7	0.4	1.0	1.3	1.7	1.0	1.6	1.2	0.7	0.7	0.4	0.4	0.6	1.0	1.3
E				0.4	0.8	1.1	0.0	0.9	1.5	2.3	2.1	2.0	0.7	0.8	0.8	0.8	0.6	0.9	1.3	1.0
A					1.0	0.9	0.8	1.3	0.6	0.5	0.7	1.3	1.1	1.2	1.7	0.5	0.8	1.2	1.3	0.7
N						0.8	1.1	1.5	1.6	1.2	2.0	0.4	0.8	0.7	0.7	0.7	0.5	0.6	1.1	0.4
Q							1.4	1.6	1.7	0.8	1.0	1.5	0.6	0.6	0.3	1.0	1.0	0.7	1.1	1.7
S								2.5	1.3	1.3	1.4	1.0	0.6	0.5	1.3	0.4	0.9	0.7	0.5	0.7
T									2.3	0.9	0.7	1.4	1.0	0.8	0.7	1.1	0.8	0.5	0.5	0.4
K										0.9	0.8	0.8	0.9	1.0	0.6	0.8	1.0	0.6	1.1	1.2
R											0.7	1.6	0.9	0.5	1.7	1.4	0.7	0.5	1.6	0.9
H												0.8	1.0	0.6	0.4	1.1	0.6	1.0	1.3	0.0
V													1.1	1.2	1.2	1.4	1.1	1.0	0.8	0.8
I														1.5	0.9	0.8	1.1	1.2	0.8	0.8
M															2.9	1.1	0.6	1.1	0.9	0.0
C																1.4	1.4	0.8	0.9	0.4
L																	1.3	1.4	0.9	1.3
F																		1.3	1.3	1.5
Y																			1.4	2.6
W																				3.1

Table 6.10: All diagonal pairs

Parallel	Observed Counts	H	300	208	120
		N		40	64
		P			20
	Observed/Expected	H	1.0	1.0	0.9
		N		1.0	1.2
		P			1.2
Antiparallel	Observed counts	H	532	408	335
		N		162	239
		P			104
	Observed/Expected	H	1.2	0.8	0.8
		N		1.2	1.1
		P			1.2
Both	Observed Counts	H	832	616	455
		N		202	303
		P			124
	Observed/Expected	H	1.1	0.9	0.8
		N		1.2	1.2
		P			1.2

Table 6.11: Diagonal pairs: counts and preferences for parallel, antiparallel, and all-beta, for amino acids grouped into three classes.

Charged pairs

In beta pairs and diagonal pairs, positively charged residues (Lys, Arg, sometimes His) associate very strongly, and for the most part significantly, with negatively charged residues (Asp, Glu). This is also true of the contacting pairs computed in chapter 4, in which secondary structure was not a filter. In $(i, i+2)$ pairs, however, there is no significant pairing; Lys shows a slight tendency not to associate with Asp and Glu, and (Arg, Glu) pairs are also disfavored.

Pairs of positively charged residues do not show the strong anti-association one might expect. They generally show negative association, and sometimes positive association. Most occurrences of beta, $(i, i+2)$, and diagonal pairs are not significant. (His, His) pairs, however, are significantly favored. Looking at the general pairs, (His, His) pairs and (Arg, Arg) pairs are favored, while (His, Arg) and (His, Lys) pairs are disfavored. Thus we do not see a clear effect of charge repulsion here. This is perhaps partly due to His not always being charged, and Arg and Lys having long, flexible sidechains with the charges at the end, which allows the same-sign charges to avoid each other.

Pairs of negatively charged residues are slightly disfavored in beta pairs, and slightly favored in $(i, i+2)$ pairs. In diagonal pairs, (Asp, Asp) and (Glu, Glu) pairs are disfavored, but (Asp, Glu) pairs are favored. None of these are significant.

Asn and Gln

Asn and Gln are polar and capable of being both hydrogen bond donors and acceptors. I expected that they would associate well with each other, and with charged residues, as well as with the polar Ser and Thr. In beta pairs, Asn and Gln do show strong significant self-association. This is not the case for $(i, i+2)$ pairs, where the association is slightly (though not significantly) disfavorable. In all contacting pairs, the self-association is favorable. There is not strong association one way or the other between Asn or Gln and the charged residues. The association that does exist changes sign in the various topological relationships examined.

Ser and Thr

In beta pairs, $(i, i+2)$ pairs, and diagonal, Ser and Thr show very strong self-association, as well as strong (Ser, Thr) association. Thr prefers association with polar residues, and disfavors hydrophobic residues as beta pair partners. In beta pairs, here is significant association for (Thr, Asn) and (Thr, Arg). In beta pairs, there is significant avoidance in the pairs (Thr, Val), (Thr, Leu), (Thr, Phe), and (Thr, Trp). Ser also tends to favor polar and disfavor hydrophobic beta pair partners, though not with the fervor of Thr.

Cys

(Cys, Cys) pairs are the most strongly favored in the table of all contacting pairs. (Cys, Cys) pairs occur in antiparallel beta pairs and antiparallel diagonal pairs. They do *not* occur in parallel beta pairs, $(i, i+2)$ pairs, or parallel diagonal pairs. This is probably due to the fact that the geometry of these topological pairs is not conducive to forming the disulfide bond that favors the (Cys, Cys) pairs in other conformations. This is the most striking example of differences in pairwise preference as a function of the particular pair topology, and provides a strong argument for distinguishing between different types of pairs in the protein structure representation.

6.3.2 Significant specific "recognition" of beta pairs

The χ^2 level of confidence for rejecting the null hypothesis of random pairing between β-strands is 100% for β_A (χ^2= 180.28 with 78 degrees of freedom) and 100% for β_P (χ^2= 92.87 with 21 degrees of freedom). Thus I agree with Lifson and Sander's result that there is statistically significant recognition.

6.3.3 Significant nonspecific recognition

The Lifson and Sander 1980 model does not consider the environment. The contingency table is therefore collapsed to two dimensions as shown in Table 6.12. Also shown are the counts predicted by the model which assumes random association of

	Counts		
	VLIFYWCM	GASTP	KRHDENQ
VLIFYWCM	5890	2202	1734
GASTP	2202	1820	1355
KRHDENQ	1734	1355	2098

	Expected counts, independence model		
	VLIFYWCM	GASTP	KRHDENQ
VLIFYWCM	4735	2591	2500
GASTP	2591	1418	1368
KRHDENQ	2500	1368	1320

	Observed/expected counts		
	VLIFYWCM	GASTP	KRHDENQ
VLIFYWCM	1.24	0.85	0.69
GASTP	0.85	1.28	0.99
KRHDENQ	0.69	0.99	1.59

	$G^2 = 2obs \times \ln(obs/exp)$		
	VLIFYWCM	GASTP	KRHDENQ
VLIFYWCM	2571	-717	-1268
GASTP	-717	909	-26
KRHDENQ	-1268	-26	1946

Table 6.12: Two-dimensional contingency table showing beta pair counts; expected counts based on a model of independent random association of pairs; likelihood ratio, observed to expected counts; G^2

amino acid type. The observed data shows a significant non-random association of amino acid class, with G^2= 1404 and 4 degrees of freedom (or accounting for symmetry, G^2 of 1202 and 3 degrees of freedom). The ratio of observed to expected counts shows that amino acids prefer to form beta pairs with amino acids of the same class. (hydrophobe with hydrophobe, neutral with neutral, and polar with neutral). Polar and hydrophobic residues show a clear preference not to be beta paired. The likelihood ratio statistic G^2 for the individual entries are also shown in Table 6.12; all of these are significantly different from 0, for one degree of freedom.

6.3.4 Solvent exposure

Table 6.13 shows counts and frequencies for beta pairs whose residues are both buried.

Buried Beta Pair Counts

	G	P	D	E	A	N	Q	S	T	K	R	H	V	I	M	C	L	F	Y	W
G	7	3	8	5	15	6	2	11	19	7	8	8	49	15	9	9	29	15	13	8
P		1	1	3	0	4	3	4	3	6	4	1	7	4	4	3	4	3	4	1
D			1	2	8	5	5	12	13	7	10	7	7	13	3	0	10	3	3	0
E				2	8	6	6	6	14	32	16	6	14	16	1	0	6	8	8	2
A					8	5	5	17	32	18	5	3	56	45	4	7	36	21	17	4
N						2	4	8	13	11	4	3	14	5	3	4	5	7	9	4
Q							2	10	15	6	9	1	11	8	3	2	10	11	13	3
S								18	34	17	9	6	38	18	5	4	21	13	13	3
T									17	17	17	16	33	28	8	4	23	9	16	3
K										6	3	4	20	20	2	6	16	9	19	5
R											1	5	20	17	4	2	13	9	10	4
H												1	14	12	0	4	10	0	5	3
V													85	97	19	14	91	53	28	10
I														35	11	15	73	31	26	8
M															1	4	13	7	9	3
C																6	11	11	8	8
L																	33	34	32	12
F																		15	14	6
Y																			11	7
W																				2

Buried Beta Pair Likelihood Ratios

	G	P	D	E	A	N	Q	S	T	K	R	H	V	I	M	C	L	F	Y	W
G	1.1	0.9	1.3	0.6	0.9	1.0	0.3	0.8	1.1	0.6	0.9	1.5	1.3	0.6	1.6	1.4	1.1	1.0	0.9	1.6
P		2.5	0.7	1.5	0.0	2.5	1.8	1.1	0.7	2.0	1.8	0.7	0.7	0.6	2.8	1.9	0.6	0.8	1.1	0.8
D			0.7	0.5	1.1	1.7	1.6	1.8	1.6	1.3	2.5	2.7	0.4	1.0	1.1	0.0	0.8	0.4	0.5	0.0
E				0.8	0.8	1.5	1.4	0.7	1.2	4.2	2.9	1.7	0.6	0.9	0.3	0.0	0.4	0.8	0.9	0.6
A					0.8	0.6	0.6	0.9	1.4	1.2	0.5	0.4	1.1	1.3	0.6	0.9	1.1	1.1	1.0	0.6
N						1.3	1.2	1.1	1.5	1.9	1.0	1.1	0.7	0.4	1.1	1.3	0.4	1.0	1.3	1.7
Q							1.2	1.4	1.6	1.0	2.0	0.4	0.6	0.6	1.0	0.6	0.7	1.4	1.8	1.2
S								2.2	1.7	1.3	0.9	1.0	0.9	0.6	0.8	0.6	0.7	0.8	0.8	0.5
T									1.4	1.0	1.4	2.1	0.6	0.8	1.0	0.4	0.6	0.4	0.8	0.4
K										1.1	0.4	0.8	0.6	0.8	0.4	1.0	0.7	0.7	1.5	1.1
R											0.3	1.3	0.8	0.9	1.0	0.5	0.7	0.9	1.1	1.2
H												0.8	0.8	1.0	0.0	1.4	0.9	0.0	0.8	1.4
V													1.5	1.2	1.1	0.7	1.2	1.2	0.7	0.7
I														1.2	0.9	1.1	1.3	1.0	0.9	0.8
M															0.8	1.4	1.1	1.1	1.4	1.4
C																3.7	0.8	1.5	1.1	3.2
L																	1.3	1.1	1.1	1.2
F																		1.8	0.9	1.1
Y																			1.5	1.3
W																				2.1

Table 6.13: Counts (top) and frequencies (bottom) for buried beta pairs.

Likelihood Ratios						
	All beta pairs			Buried beta pairs		
hydrophobic	1.1	1.0	0.9	1.1	0.9	0.8
neutral		1.0	1.1		0.7	1.5
polar			1.2			1.3

Table 6.14: R_{ij}^{β} for three classes.

Table 6.14 shows R_{ij}^{β} for three classes. Residues that are buried seem to show a more definite recognition between classes, but this may also be partly due to the reduced number of counts. A χ^2 analysis is pending.

6.3.5 Five-dimensional contingency table

Table 6.15 shows the five-dimensional contingency table generated from all beta pairs in the set of proteins. The counts range from 3 (parallel, exposed hydrophobe next to exposed neutral) to 3074 (antiparallel, buried hydrophobe next to buried hydrophobe).

The attributes of each pair are defined as follows:

- A_1, A_2: the amino acid type (grouped into three hydrophobicity classes) of the two members of the beta pair.
- E_1, E_2: the solvent exposure of each of the two members of the beta pair.
- D: the strand direction of the beta pair.

6.3.6 Nonrandom association of exposure

Table 6.16 summarizes the counts by reporting the three-dimensional marginal totals obtained by summing over amino acid type. The table shows that the environment (factor) variables are not independent of one another. Numbers in parentheses show expected counts assuming random pairing according to the two-dimensional table margin totals. A model of the five-dimensional contingency table incorporating the two-factor term corresponding to the exposures of each residue reduces the G^2 statistic

			Buried			Exposed		
			Hyd	Neu	Pol	Hyd	Neu	Pol
Antipar.	Buried	Hyd	3074	1295	916	484	180	349
		Neu	1295	856	694	133	262	225
		Pol	916	694	858	72	63	298
	Exposed	Hyd	484	133	72	204	116	129
		Neu	180	262	63	116	182	211
		Pol	349	225	298	129	211	390
Parallel	Buried	Hyd	1436	420	162	102	29	85
		Neu	420	162	93	26	40	45
		Pol	162	93	78	8	11	58
	Exposed	Hyd	102	26	8	4	3	13
		Neu	29	40	11	3	16	13
		Pol	85	45	58	13	13	60

Table 6.15: Five-dimensional contingency table.

by 1339 over the single-term model, with a change of one degree of freedom, which is highly significant. This observation, along with the knowledge that each amino acid shows a preference for either buried or exposed positions, suggests that some of Lifson and Sander's correlation between amino acids is likely to be due to the nonrandom association of exposure environment.

Loglinear models can be created for the full five-dimensional contingency table to illustrate this point. I examined two model hierarchies; they add terms in different orders. Starting with all single margins, and adding two-way exposure, then direction paired terms, I have the models shown in Table 6.17. Another possible order of models is shown in Table 6.18; here I start with all single margins, then add the two-way exposure/direction margins $[E_1D][E_2D]$, then the two-way exposure margin $[E_1E_2]$, then the three-way term $[E_1E_2D]$.

Note that while the order of adding terms here does make a difference in the amount that the goodness of fit statistic is increased from one model to the next, the largest effect is due to the two-way term $[E_1E_2]$.

Exposure preference accounts for 1/3 to 1/2 of the variance

I ask how the solvent exposure might affect the pairwise recognition results. By separating the buried and exposed counts, how much of the association in the Lifson

Antiparallel Beta Pairs			
	Buried	Exposed	Total
Buried	10,598	2,066	12,664
	9,768	*2,896*	*12,664*
Exposed	2,066	1,688	3,754
	2,896	*858*	*3,754*
Total	12,664	3,754	16,418

Parallel Beta Pairs			
	Buried	Exposed	Total
Buried	3,026	404	3,430
	2,962	*468*	
Exposed	404	138	542
	468	*74*	
Total	3430	542	3972

All Beta Pairs			
	Buried	Exposed	Total
Buried	13,624	2,470	
Exposed	2,470	1,826	

Table 6.16: $[E_1E_2D]$ margin totals.

Added margins	G^2	df	ΔG^2	Δ(df)	interpretation
$[A_1][A_2][E_1][E_2][D]$	5940	64			single margins maintained
$[E_1E_2]$	4601	63	1339	1	two-way exposure
$[E_1D]$,$[E_2D]$	4319	61	282	2	two-way exposure/direction
$[E_1E_2D]$	4301	60	18	1	three-way exp/dir

Table 6.17: Model hierarchy for the environment variable interactions

Added margins	G^2	df	ΔG^2	Δ(df)	margins maintained
$[A_1][A_2][E_1][E_2][D]$	5940	64			single margins maintained
$[E_1D]$,$[E_2D]$	5586	62	354	2	two-way exp/dir
$[E_1E_2]$	4319	61	1267	1	two-way exp
$[E_1E_2D]$	4301	60	18	1	three-way exp/dir

Table 6.18: Model hierarchy for the environment variable interactions; alternate ordering

Added margins	G^2	df	ΔG^2	Δ(df)	interpretation
$[A_1][A_2][E_1][E_2][D]$	5940	64			single margins
$[E_1E_2D]$	4301	60	1639	4	full environment
$[A_1E_1][A_2E_2]$	2235	56	2066	4	residue's own exposure
$[A_1E_2][A_2E_1]$	2149	52	86	4	other residue's exposure
$[A_1D][A_2D]$	1704	48	445	4	direction
$[A_1E_1E_2][A_2E_1E_2]$	1687	44	17	4	3-way exposure
$[A_1E_1E_2D][A_2E_1E_2D]$	1403	32	284	12	full-way environment

Table 6.19: Model hierarchy comparing exposure and strand direction. Each model in the hierarchy is described by the maximal margins added.

Added margins	G^2	df	ΔG^2	Δ(df)	interpretation
$[A_1][A_2][E_1][E_2][D]$	5940	64			single margins
$[E_1E_2D]$	4301	60	1639	4	full environment
$[A_1D][A_2D]$	3681	56	620	4	direction
$[A_1E_1][A_2E_2]$	1772	52	1909	4	residue's own exposure
$[A_1E_2][A_2E_1]$	1704	48	68	4	other residue's exposure
$[A_1E_1E_2][A_2E_1E_2]$	1687	44	17	4	3-way exposure
$[A_1E_1E_2D][A_2E_1E_2D]$	1403	32	284	12	full-way environment

Table 6.20: Adding terms in a different order.

and Sander model can I explain? For this analysis, I assume that the exposure and direction variables are the independent variables, and start with a model that includes all their margins, model $[A_1][A_2][E_1E_2D]$.

Results are shown in Table 6.19. The main result is that the residue's own exposure accounts for much (one-third to one-half) of the variance in model $[A_1][A_2][E_1E_2D]$.

Model $[A_1E_1E_2D][A_2E_1E_2D]$ treats each environment table separately. Accounting for symmetry, the model has G^2 of 701 with 9 degrees of freedom. This is to be compared with the Lifson and Sander model, which does not take environment into account, and has a G^2 of 1202 with 3 degrees of freedom. Both models show significant nonrandom association, but model $[A_1E_1E_2D][A_2E_1E_2D]$ shows less.

In Table 6.20, I check to see whether the contribution of terms to the G^2 statistic is affected by the order in which they are added to the models. Clearly, regardless of the order in which terms are added, the residue's own exposure is the largest contributor to the non-randomness in the models.

Counts			
	Hyd.	Neu.	Pol.
Hydrophobic	1066	466	292
Neutral	466	176	85
Polar	292	85	38

Observed/Expected			
	Hyd.	Neu.	Pol.
Hydrophobic	0.95	1.04	<u>1.14</u>
Neutral		0.99	0.84
Polar			<u>0.65</u>

Table 6.21: Counts and likelihood ratios for $i, j+1$ pairs, with amino acids grouped by hydrophobicity class. Statistically significant cells (P< 0.05) are underlined in the likelihood ratio table.

6.3.7 Unexpected correlation in buried $(i, j+1)$ pairs

I considered buried $(i, j+1)$ pairs, expecting this to be a baseline with no correlation, because the side chains are not in contact on opposite sides of the sheet and on different beta strands. However, I found significant correlation.

For amino acids grouped into three hydrophobicity classes, $\chi^2 = 18.58$ with three degrees of freedom, and P= 3.3×10^{-4}. Table 6.21 shows counts and likelihood ratios, with significant cells (P< 0.05) underlined. The two significant cells in the table are polar-polar and polar-hydrophobic. Polar residues disfavor polar residues in this relationship (opposite sides of a buried beta sheet). Polar residues favor hydrophobic residues. This suggests a residual amphipathicity in sheets that are buried.

Does this change with a more stringent definition of buried? Perhaps we're actually still looking at amphipathic sheets because of a definition that allows some exposed residues to masquerade as buried residues. Table 6.22 shows counts and likelihood ratios for a more stringent definition of buried: relative exposure must be less than 5%, rather than the 20% used in Table 6.21. There are 1243 total pairs, as opposed to 2123 at the 20% level. Only the polar-polar cell is significant at the 0.05 level ($\chi^2 = 5.44$; P= 0.02). The overall table has $\chi^2 = 8.6$ with three degrees of freedom, which is significant at the 0.05 level, P= .035. It appears that polar residues tend to appear less than expected in the $(i, j+1)$ relationship in very buried beta

Counts			
	Hyd.	Neu.	Pol.
Hydrophobic	660	287	125
Neutral	287	122	41
Polar	125	41	8

Observed/Expected			
	Hyd.	Neu.	Pol.
Hydrophobic	0.97	1.01	1.14
Neutral		1.02	0.89
Polar			<u>0.45</u>

Table 6.22: Counts and likelihood ratios for $i, j+1$ pairs, with amino acids grouped by hydrophobicity class. Statistically significant cells are underlined in the likelihood ratio table.

sheets.

I grouped the amino acids into eight groups: positively charged (KRH), negatively charged (DE), aromatic (FYW), hbond donors (ST), hbond donors and acceptors (NQ), hydrophobic (VLI), small (GAP), and sulfur-containing (CM). With the amino acids grouped into these eight groups, $\chi^2 = 68.8$ with 28 degrees of freedom, and P= 2.7×10^{-5}. Table 6.23 shows counts and likelihood ratios, with significant cells in bold face. The significantly disfavored opposite-sheet-side pairs are (1) negatively charged with negatively charged, (2) aromatic with aromatic, (3) hydrogen bond donor with hydrogen bond donor/acceptor, and (4) sulfur-containing with hydrophobic. The significantly favored opposite-sheet-side pairs are (1) hydrogen bond donor/acceptor with aromatic, (2) hydrogen bond donor with aromatic, and (3) sulfur-containing with sulfur-containing.

For the 20 by 20 table, $\chi^2 = 229.6$ with 190 degrees of freedom, and P= 0.026. Table 6.24 shows counts and likelihood ratios, with significant cells (P< 0.05) underlined. For all the significant cells, the co-occurence of those amino acid pairs on opposite sides of the sheet is *greater* than would be expected by random association. These pairs are GG, CG, FD, IE, RN, YQ, FS, FT, VV, MM, YC.

There are several possible explanations for this nonrandom association. One is that the environments are somehow different on the two sides of the secondary struc-

Counts								
	KRH	DE	FYW	NQ	ST	VLI	GAP	CM
KRH	6	4	26	10	12	80	24	10
DE	4	0	24	1	10	57	18	8
FYW	26	24	46	27	59	157	60	33
NQ	10	1	27	2	5	53	16	7
ST	12	10	59	5	28	138	40	20
VLI	80	57	157	53	138	534	167	44
GAP	24	18	60	16	40	167	68	22
CM	10	8	33	7	20	44	22	18

Observed/Expected								
	KRH	DE	FYW	NQ	ST	VLI	GAP	CM
KRH	0.60	0.57	1.04	1.43	0.66	1.12	1.00	1.06
DE		0.00	1.35	0.20	0.78	1.13	1.05	1.20
FYW			0.73	1.53	1.30	0.88	0.99	1.40
NQ				0.41	0.39	1.06	0.95	1.06
ST					0.85	1.07	0.92	1.17
VLI						1.05	0.97	0.65
GAP							1.17	0.97
CM								2.03

Table 6.23: Counts and likelihood ratios for $i, j+1$ pairs, with amino acids grouped into eight classes. Statistically significant cells are underlined in the likelihood ratio table.

Counts																				
	G	P	D	E	A	N	Q	S	T	K	R	H	V	I	M	C	L	F	Y	W
G	16	1	5	2	12	2	3	7	8	2	1	3	30	21	5	9	21	13	8	1
P	1	0	0	0	3	2	0	2	4	1	1	1	3	4	1	0	4	2	0	0
D	5	0	0	0	4	0	1	2	4	0	0	1	9	5	1	1	9	9	4	1
E	2	0	0	0	7	0	0	1	3	2	0	1	5	16	4	2	13	6	2	2
A	12	3	4	7	20	7	2	8	11	7	8	0	40	24	3	4	20	16	13	7
N	2	2	0	0	7	0	0	3	1	1	4	3	9	10	4	1	7	9	4	1
Q	3	0	1	0	2	0	2	0	1	0	1	1	9	7	1	1	11	5	6	2
S	7	2	2	1	8	3	0	6	8	2	1	1	21	16	5	4	19	18	9	3
T	8	4	4	3	11	1	1	8	6	2	3	3	29	26	7	4	27	21	5	3
K	2	1	0	2	7	1	0	2	2	0	1	0	11	5	2	3	9	5	5	0
R	1	1	0	0	8	4	1	1	3	1	0	2	12	8	1	1	10	4	2	3
H	3	1	1	1	0	3	1	1	3	0	2	0	14	7	2	1	4	5	1	1
V	30	3	9	5	40	9	9	21	29	11	12	14	102	68	12	7	58	25	17	9
I	21	4	5	16	24	10	7	16	26	5	8	7	68	46	6	5	45	20	23	12
M	5	1	1	4	3	4	1	5	7	2	1	2	12	6	6	4	5	8	5	3
C	9	0	1	2	4	1	1	4	4	3	1	1	7	5	4	4	9	2	12	3
L	21	4	9	13	20	7	11	19	27	9	10	4	58	45	5	9	44	20	18	13
F	13	2	9	6	16	9	5	18	21	5	4	5	25	20	8	2	20	12	7	2
Y	8	0	4	2	13	4	6	9	5	5	2	1	17	23	5	12	18	7	8	3
W	1	0	1	2	7	1	2	3	3	0	3	1	9	12	3	3	13	2	3	2

Observed/Expected																				
	G	P	D	E	A	N	Q	S	T	K	R	H	V	I	M	C	L	F	Y	W
G	1.6	0.6	1.6	0.5	1.0	0.5	1.0	0.9	0.8	0.6	0.3	1.0	1.1	1.0	1.0	2.0	1.0	1.1	0.9	0.3
P		0.0	0.0	0.0	1.4	3.0	0.0	1.5	2.3	1.8	1.6	2.0	0.6	1.1	1.2	0.0	1.1	1.0	0.0	0.0
D			0.0	0.0	1.0	0.0	1.0	0.8	1.2	0.0	0.0	1.0	1.0	0.7	0.6	0.7	1.3	2.3	1.4	0.8
E				0.0	1.5	0.0	0.0	0.3	0.8	1.6	0.0	0.9	0.5	1.9	2.1	1.2	1.6	1.3	0.6	1.3
A					1.3	1.4	0.5	0.8	0.9	1.7	1.7	0.0	1.1	0.9	0.5	0.7	0.8	1.1	1.2	1.4
N						0.0	0.0	1.0	0.3	0.8	2.8	2.6	0.8	1.2	2.0	0.6	0.8	1.9	1.2	0.6
Q							2.1	0.0	0.3	0.0	0.9	1.1	1.0	1.1	0.7	0.7	1.7	1.3	2.2	1.6
S								1.0	1.0	0.8	0.4	0.4	0.9	0.9	1.3	1.1	1.1	1.9	1.3	0.9
T									0.6	0.6	0.8	1.0	1.0	1.2	1.4	0.9	1.2	1.7	0.6	0.7
K										0.0	0.8	0.0	1.2	0.7	1.2	2.0	1.3	1.2	1.7	0.0
R											0.0	1.9	1.2	1.0	0.6	0.6	1.3	0.9	0.6	2.0
H												0.0	1.7	1.1	1.4	0.8	0.6	1.4	0.4	0.8
V													1.3	1.1	0.9	0.6	1.0	0.7	0.7	0.8
I														1.0	0.6	0.5	1.0	0.8	1.2	1.3
M															2.5	1.8	0.5	1.3	1.2	1.5
C																2.0	1.0	0.4	3.0	1.6
L																	1.0	0.8	1.0	1.5
F																		0.8	0.7	0.4
Y																			1.0	0.8
W																				1.2

Table 6.24: Counts and likelihood ratios for $i, j + 1$ pairs. Statistically significant cells (P< 0.05)are underlined in the likelihood ratio table.

ture element, in a way that is not being picked out just with secondary structure and DSSP solvent exposure. Different environments would imply different "singleton" amino acid preferences, and the resultant segregation would lead to an observed pairwise effect, if the environments were not carefully separated in the counting.

Another explanation is that local sequence composition is a strong factor in determining local secondary structure. The $(i, j+1)$ interaction is a combination of an $(i, i+1)$ effect and a (i, j) effect. As we've seen earlier in this chapter, (i, j) pairs show self-association between hydrophobicity category. Moreover, buried beta $(i, i+1)$ pairs, even though they are not in contact, show anti-association of hydrophobicity type.

6.4 Conclusions

This work explores the idea of incorporating knowledge about the topological relationship between core element positions in protein structural models for threading algorithms. Other work has used information about the distance between pairs, and the secondary structure of a single residue position.

The side chains of amino acid residues in proteins interact in complex ways with their spatial neighbors. Each side chain is in contact with several others. Many pseudo-energy functions that evaluate the quality of a sequence/structure match sum pairwise numbers for each residue pair without regard to the environment of a residue. These results show, however, that the frequencies of pairs of amino acid residues in a very specific topological relation within the protein can vary significantly depending on the local protein topology and environment.

My results also show the importance of considering the environment in compiling statistics for use in prediction algorithms. For example, I've shown that the parallel/antiparallel frequency counts are greatly affected by solvent exposure. In general, care should be taken to be sure that the environments are similar for the categories being examined. For example, when I take solvent exposure out of the picture, and consider only buried residues, it is clear that the hydrophobic residues that

are branched at the beta carbon dominate all other residues in their beta-structure preference. In addition, it is important to properly separate the contributions of various environmental factors to the pseudopotential; threading algorithms in particular enable this to be done in a rational way because the protein models contain all the necessary environmental information.

The $N^{\beta,\text{touching}}$ and $f^{\beta,\text{touching}}$ results suggests that using frequencies derived for topologically related residues may be valid regardless of whether the residues are in contact. Topological relation may be as important as physical adjacency. Moreover, residues that are topological neighbors may jointly influence the structure even though they are not in contact: consider two small residues, or the antiparallel conformation where every other pair is quite distant from each other.

It might be useful to use only statistically significant pairwise interactions in evaluation functions, to avoid the problem of low sample size.

The overriding result here is that the interaction of solvent exposure and amino acid hydrophobicity are key. Strand direction has a significant but much smaller effect.

Chapter 7

Secondary Structure Prediction

7.1 Introduction

In the previous chapters I have shown that solvent exposure and hydrophobicity are of primary importance in protein structure. I have also shown that some buried structural elements are amphipathic. Clearly, patterns of hydrophobicity are important in determining the protein's fold.

I turn now to the question of how the patterns of hydrophobicity might be used in protein structure prediction. In this chapter I show that hydrophobicity patterns in a protein sequence can be exploited to improve secondary structure prediction. The hydrophobic patterns are computed as two numbers indicating the amphipathic strength of a stretch of residues along the sequence. These two numbers are used to augment the inputs to a neural network that predicts secondary structure. The addition of this hydrophobicity pattern information provides a small but significant improvement in the performance of the neural network.

This chapter includes several other items of interest. I describe the methodology for determining the significance of performance improvement in a cross-validated study. The neural network weights are examined and displayed. Finally, I explore in some detail the characteristics of a representation of hydrophobicity patterns.

7.2 Related Work

There are many different secondary structure prediction algorithms. Some take advantage of hydrophobicity patterns. Some use neural networks. There are also several ways that hydrophobicity patterns have been represented.

In this section, I discuss other work using hydrophobicity patterns in secondary structure predictions. I then summarize related work in neural networks for predicting secondary structure. I conclude the section by discussing representations of hydrophobicity patterns.

7.2.1 Hydrophobicity patterns in other secondary structure prediction methods

The most commonly used secondary structure predictions methods are the Chou-Fasman for globular proteins [Chou and Fasman, 1978], and the helical wheel for finding trans-membrane helices [Schiffer and Edmundson, 1967]. There are many other approaches, some of which I discuss here.

The Chou-Fasman secondary structure prediction considers secondary structure propensity, but does not look for patterns of hydrophobicity [Chou and Fasman, 1978]. (Secondary structure propensity is the likelihood ratio P(SA)/P(S)P(A), where S is the secondary structure, A is the amino acid, and P(S), P(A) and P(SA) are the probabilities of occurrence of S, A, and A and S jointly, respectively.) The method looks for strong secondary-structure forming groups of residues and grows the strands and helices outward from these nucleii, based only on the secondary structure propensity of each amino acid.

The GOR method [Garnier *et al.*, 1978] also does not use hydrophobicity patterns. The method considers a window of residues and predicts the secondary structure of the central window, based on the first-order contribution of each residue type at each position relative to the central position. The GOR method is very closely related to the single-layer neural network approach with bit representation that I use as a control. The weights in the learned network (see Figure 7-12) correspond to the GOR

parameters computed for each residue in the window.

The helical wheel approach looks for changes in hydrophobicity with a period equal to that expected for alpha helices, 3.6 residues [Schiffer and Edmundson, 1967]; this has been implemented in a computer program [Kyte and Doolittle, 1982]. This has been most often used for finding transmembrane helices.

Eisenberg and colleagues defined the hydrophobic moment to indicate the degree of alternation of hydrophobic and hydrophilic residues at a given frequency in a protein sequence. They used the moment along with average hydrophobicity to distinguish between types of alpha helices (globular, membrane, and membrane-surface-seeking) [Eisenberg *et al.*, 1982, Eisenberg *et al.*, 1984a]. Beta-strand and other helical structures were also analyzed [Eisenberg *et al.*, 1984b]. The hydrophobic moment was used with some variations by Finer-Moore and Stroud in a secondary structure prediction for the acetylcholine receptor [Finer-Moore and Stroud, 1984].

Neural nets with hidden units have the representational power to exploit hydrophobicity patterns without being given any explicit information other than the amino acid sequence. However, there is evidence, discussed in section 7.2.2, that neural nets do not utilize these patterns to their advantage. Kneller and colleagues provided neural networks with explicit information about hydrophobic moments [Kneller *et al.*, 1990].

Lim's complex prediction rules make explicit use of patterns of amphiphilicity [Lim, 1974]. The rules include amphipathicity in strands and helices. There exist computer implementations of his approach [Lenstra, 1977, Nishikawa, 1983]. The rules were built by hand, and the effect of their interactions is not clear.

While not confined exclusively to secondary structure prediction, Taylor's template-based methods can be used for such applications, and they easily accommodate the expression of hydrophobicity patterns [Taylor, 1989]. Again, the patterns tend to be built by hand. This approach remains relatively unproved for secondary structure prediction.

The pattern-matching approach to secondary structure prediction of Cohen and colleagues also has the power to represent hydrophobicity patterns [Cohen *et al.*,

1983]. One of the patterns they use looks for a hydrophilic side on a helix.

Ross King's work on inductive learning to find patterns useful to predict secondary structure included the automatic discovery of patterns which represent amphipathicity of secondary structures [King, 1988].

7.2.2 Neural Nets for Predicting Secondary Structure

Neural networks have been used by several groups to predict secondary structure [Presnell and Cohen, 1993, Stolorz *et al.*, 1991]. The networks appear to perform as well as any other secondary structure prediction method. I find them particularly useful for this work because it is straightforward to manipulate the type of information presented to the network.

In the late 1980's there were several applications of neural nets to secondary structure prediction [Qian and Sejnowski, 1988, Holley and Karplus, 1989]. In each case, the inputs were encoded using a bit representation, one input per type of amino acid. A local window in the sequence was coded in this way as input to the network. The output was three units, representing alpha, beta and coil (other) structure. Qian and Sejnowski tried various representations, using physico-chemical parameters, including the Garnier propensities [Qian and Sejnowski, 1988]. But none of these improved performance over the 21-bit representation.

Maclin and Shavlik used a neural network to improve the performance of the Chou-Fasman secondary structure prediction algorithm [Maclin and Shavlik, 1991]. They translated a finite state automaton implementing the Chou-Fasman algorithm into starting weights for a neural network. The net was then trained and the final result was a better prediction than either Chou-Fasman or other neural net approaches.

Rost and Sander combined information from aligned sequences to obtain improved secondary structure prediction [Rost and Sander, 1993a].

Neural networks have been used in other closely related ways. McGregor and colleagues trained a neural network to predict beta turns [McGregor *et al.*, 1989]. Several groups have used neural networks to predict protein family or folding class [Dubchak *et al.*, , Metfessel and others, 1993, Ferran and Ferrara, 1992] or restricted predic-

tions to proteins within a given class [Kneller *et al.*, 1990, Rost and Sander, 1993b]. However, Rost and Sander found that separating secondary structure prediction into a class-prediction stage followed by a structure prediction stage gave no added advantage [Rost and Sander, 1993b].

Several of the neural network results are particularly relevant to my work.

Hidden units

Qian and Sejnowski found that a network with no hidden units performed as well as a network with up to 40 hidden units [Qian and Sejnowski, 1988]. This has implications about the usefulness of explicitly providing to the network information about patterns of hydrophobicity.

There are some problems that can only be solved with a neural network that has hidden units. One example is the parity problem, which requires a 0 output if an even number of input units are equal to 1 (the rest are 0), and a 1 output otherwise. Amphipathicity patterns, like parity, can only be represented by a network with hidden units. Successfully dealing with amphipathicity successfully requires recognizing patterns of hydrophobicity, regardless of how they're shifted along the input window. For example, we might want an output node to be on whenever either BEBEB and EBEBE (where E is exposed and B is buried) were inputs, but not when EEEEE is the input.

Even a network with hidden units can have trouble finding higher-order solutions like the one required for parity or recognizing offset hydrophobicity patterns as belonging to the same class. Neural networks are notorious for preferring lower-order solutions that are easier to find in the learning process. Therefore, while the topology of the Qian and Sejnowski neural network may have been capable of representing a solution that made use of the hidden units to advantage, it appears that such a solution was not found. In this chapter, I provide the network with additional inputs to represent hydrophobicity patterns, thus turning a high-order problem into a linear recognition problem. I show that this improves network performance.

Hydrophobic moments

In work very similar to that described in this chapter. Kneller and colleagues added two units representing hydrophobic moment to the input stage of a neural network [Kneller *et al.*, 1990]. They found a small improvement in the performance of the neural network. However, their results are inconclusive for several reasons: (1) they report the improvement as being 1%; however, they have rounded to the nearest per cent in reporting the results; (2) they do not perform cross-validation but instead have a single test set; and (3) they do not show statistical significance. Moreover, their definition of hydrophobic moment is slightly different than mine: I take the maximum over several shifted subwindows that cover the residue in question.

7.2.3 Periodic features in sequences

Fourier transforms have been used to analyze periodic protein sequence features such as the distributions of charged and apolar residues in coiled-coil proteins [McLachlan and Karn, 1983]. Eisenberg and colleagues used Fourier transforms to examine hydrophobicity patterns in protein sequences.

The Eisenberg group defines the hydrophobic moment, given the 3D structure, as

$$\mu_s = \sum_{n=1}^{N} H_n s_n,$$

where H_n is the numerical hydrophobicity of the nth residue and s_n is a unit vector in the direction from the nucleus of the α carbon toward the geometric center of the side chain. If the sequence and not the structure is known, the hydrophobic moment is the magnitude of the Fourier transform of the hydrophobicity pattern along the protein sequence. The moment is computed as

$$\mu(\delta) = \left| \sum_{n=1}^{N} H_n e^{i\delta n} \right|,$$

where $\delta = 2\pi/m$ (m is the number of residues per turn).

Finer-Moore and Stroud [Finer-Moore and Stroud, 1984] modify the definition

by subtracting from H_n the average hydrophobicity for the n residues, $\overline{H}$. This has the effect of removing the origin peak at frequency 0 from the Fourier spectrum. In addition, they scaled the moment by the mean value of moments computed for the same amino acids randomly arranged in the window. From examination of two dimensional plots of moment as a function of sequence and frequency, the authors made secondary structure predictions for an acetylcholine receptor.

Cornette and colleagues used the hydrophobic moment to optimize the hydrophobicity scale based on its ability to distinguish alpha helical structure in proteins [Cornette *et al.*, 1987]. They do this by maximizing the "amphipathic index," the fraction of the total Fourier spectrum area that is under the peak corresponding to the alpha helix frequency. The amphipathic index (AI) is defined as

$$AI[P(\omega)] = \frac{\frac{1}{25^\circ}\int_{85^\circ}^{110^\circ} P(\omega)\mathrm{d}\omega}{\frac{1}{180^\circ}\int_{0^\circ}^{180^\circ} P(\omega)\mathrm{d}\omega}.$$

$P(\omega)$ is the power spectrum, or square of the hydrophic moment, and is defined as

$$P(\omega) = \left|\sum_{n=1}^{N}(H_n - \overline{H})e^{in\omega}\right|^2$$

7.3 A representation for hydrophobicity patterns

What representation of patterns in hydrophobicity along the protein sequence would be useful for a neural network approach? I wanted numbers that describe the degree to which a local region of sequence shows an alternating hydrophobic–hydrophilic pattern, with a period equal to the period of the secondary structure of interest (2 for beta and 3.6 for alpha).

7.3.1 I_α and I_β: maximum hydrophobic moments

I chose a representation based on the hydrophobic moment, which has the desired properties. It is directly related to the Fourier transform and determines the degree to which the alpha or beta frequency is present in the hydrophobicity signal. I use the hydrophobic moment in which the hydrophobicity is defined relative to the sequence window's average hydrophobicity [Finer-Moore and Stroud, 1984]. In addition, I compute the "per-residue" hydrophobic moment, dividing by the window length:

$$\mu(\omega, j, L) = \frac{2}{L} \left| \sum_{k=j+D}^{j-D} (h_k - h_a) \mathrm{e}^{2\pi i k \omega} \right|^2 ,$$

where L is the length of the window; D is the half width of the window, $D = (L-1)/2$; h_k is the hydrophobicity index of the k'th residue in the window; h_a is the average hydrophobicity number in the window; j is the position of the window's central residue in the protein sequence.

For hydrophobicity numbers, I use Ponnuswamy's hydrophobicity scale [Ponnuswamy *et al.*, 1980]. I chose a window size, L, of 5 for beta and 9 for alpha structure.

Because a residue can be near the end of a piece of secondary structure, I computed the hydrophobic moment μ over a set of five different overlapping windows containing the residue of interest. This scheme is shown in Figure 7-1. The 13-residue window shown at the top of the diagram represents the local sequence window. The secondary structure prediction will be made for the residue at the center of this window. There are five different subwindows over which μ_β is computed, and five different subwindows over which μ_α is computed. In each case, the maximum μ over these five windows is taken as the overall hydrophobic moment for input to the neural network. These two values are the beta moment

$$I_\beta = max_{j \in \{r-2,\ldots,r+2\}} \quad \mu\left(\frac{1}{2}, j, 5\right),$$

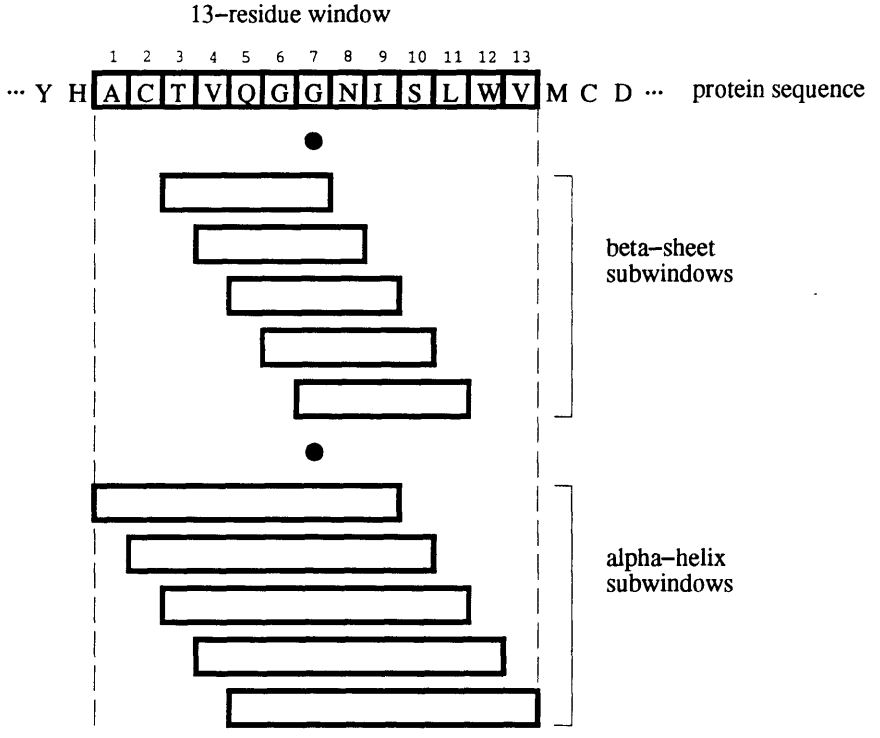


Figure 7-1: Subwindows for hydrophobicity patterns. A 13-residue window on the amino acid sequence is examined. To compute I_β, the hydrophobic moment μ_β is computed in the five different subwindows shown in the diagram. The maximum value in these five subwindows is taken as I_β. I_α is computed similarly.

	I_β ave.	I_α ave.	$I_\beta - I_\alpha$ ave.
beta	2.60	1.73	0.87
alpha	1.53	2.87	−1.34
coil	1.68	1.72	−0.04

Table 7.1: I_β and I_α for 55 proteins.

and the alpha moment

$$I_\alpha = max_{j\in\{r-2,\ldots,r+2\}} \quad \mu\left(\frac{1}{3.6}, j, 5\right),$$

where r is the index of the central residue in the 13-residue window.

7.3.2 Characterization of I_α and I_β

Before feeding these numbers to the neural networks, I characterized I_α and I_β for a number of known-structure proteins. I worked with a set of 55 single-domain, monomeric proteins, as defined in the Appendix section B.2.4. This set of proteins avoids the complexities that might be introduced by multimeric and multidomain proteins.

I_α and I_β computed for this set of proteins is shown in Figure 7-2. There is extensive overlap between secondary structure classes. However, it can be seen from the scatter plots that compared to beta residues, alpha residues tend to have higher I_α and lower I_β. The histograms of the difference between the moments are also slightly biased. The average values of $I_\beta - I_\alpha$ are −1.34 for alpha residues, 0.87 for beta residues, and an intermediate −0.04 for coil residues (Table 7.1). This ordering is just what we'd expect: I_β is on average higher than I_α for beta structures, and lower for alpha structures. From the figure, it can be seen that I_β and I_α alone are not sufficient to discriminate between secondary structures, but they do provide some information that might be useful in a neural network.

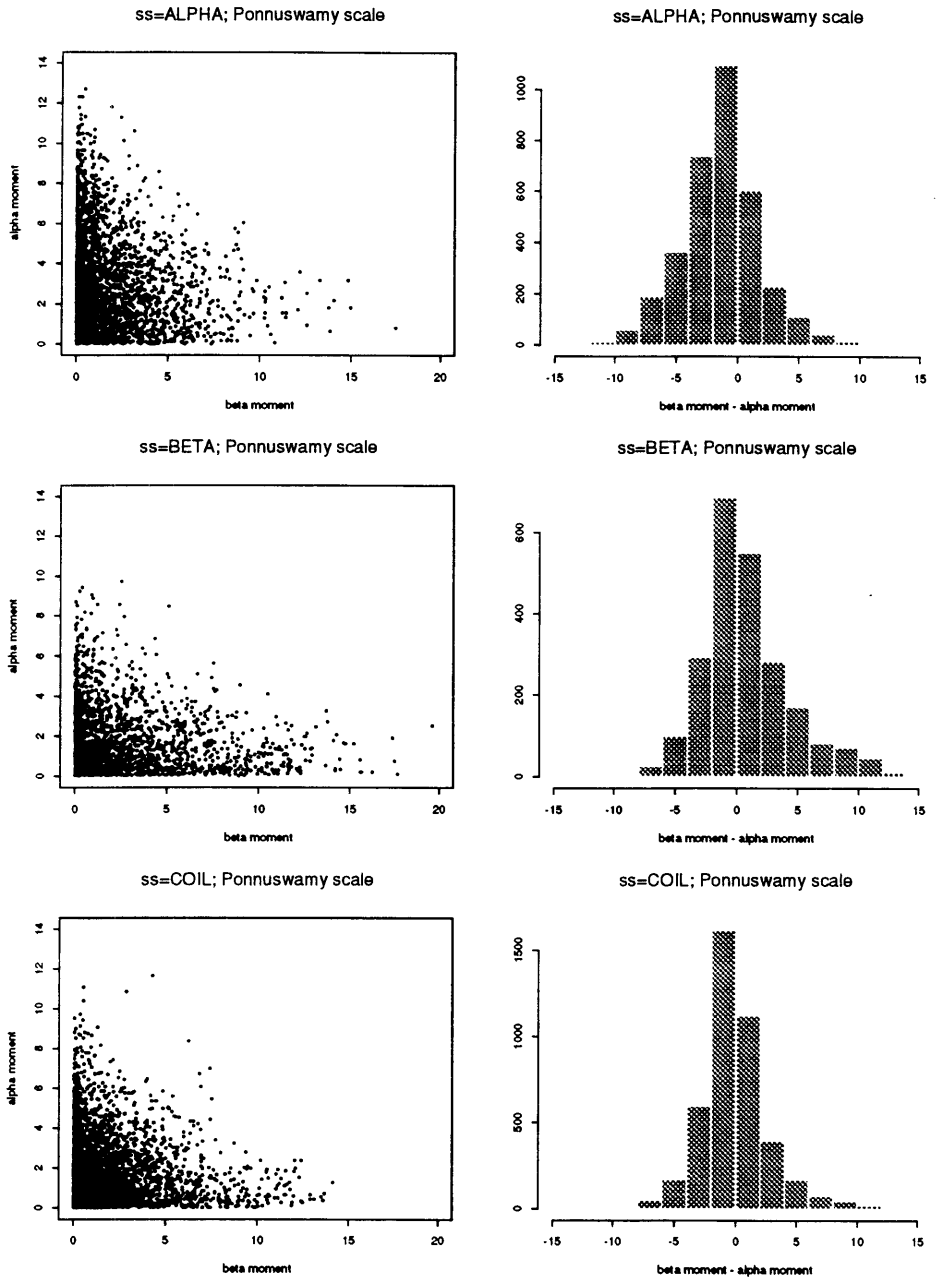


Figure 7-2: Plots of alpha and beta moments for a set of 56 proteins. The left hand column contains scatter plots of the alpha moment against the beta moment. The right hand column shows histograms of the difference between the beta moment and alpha moment. The residues are separated into alpha (top), beta (middle), and coil structure (bottom).

	Structure of residues chosen by the					
	beta rule			alpha rule		
θ	beta	alpha	coil	beta	alpha	coil
1	2340	3488	4296	2340	3488	4296
2	1334	1484	2025	1330	2656	2492
3	965	920	1239	790	1975	1419
4	731	573	795	437	1387	765
5	558	385	532	218	927	387
6	427	253	351	120	586	199
7	311	156	233	58	381	93
8	249	96	148	27	222	47
9	178	65	108	12	110	23
10	133	35	74	4	57	10
11	102	26	50	0	29	4
12	65	16	32	0	9	2
13	37	11	13	0	3	0
14	21	7	5	0	0	0
15	17	4	1	0	0	0
16	9	1	0			
17	5	1	0			
18	4	1	0			
19	1	0	0			
20	1	0	0			
21	0	0	0			

Table 7.2: Decision rule performance based on alpha and beta moments. θ is the hydrophobic moment threshold. The numbers in the table indicate how many residues of each type of secondary structure are predicted by the decision rules to have alpha or beta structure.

7.3.3 Decision rule based on I_α and I_β

One way to examine the predictive power of I_α and I_β is to use each in a decision rule for predicting secondary structure, then examine the performance of this rule on the known-structure proteins. A decision rule for beta structure is "If $I_\beta \geq \theta$, then predict beta." θ is an arbitrary threshold. Similarly, an alpha moment decision rule is "If $I_\alpha \geq \theta$, then predict alpha." Table 7.2 shows the operation of these rules on the 56-protein database.

The performance of a decision rule can be characterized by a four-element table describing the number of true positives (TN), false positives (FP), true negatives (TN),

		correct	
		Beta	Not beta
predicted	Beta	TP	FP
	Not beta	FN	TN

Table 7.3: Definition of true and false positives and negatives.

and false negatives (FN). This is shown for the case of beta structure in Figure 7.3.

Each decision rule can be described in terms of its sensitivity and specificity. Sensitivity is defined as the fraction of all beta residues that are picked up by the rule (TP/(TP+FN)). Specificity is defined as the fraction of all residues labeled beta by the rule which are in fact beta residues (TP/(TP+FP)). Figures 7-4 and 7-3 plot the sensitivity vs. specificity for the alpha and beta moments.

Another way to summarize the predicted vs. correct tables of counts for a decision rule, as a function of decision threshold, is to draw a receiver-operator characteristic (ROC) curve. Figure 7-5 plots the ROC curves for the alpha and beta moments. Along the vertical axis is the hit rate, the chances of getting an above-threshold moment given the residue does in fact have that secondary structure (TP/(TP+FN)); this is the same as sensitivity. Along the horizontal axis is the false positive rate, the chances of getting an above-threshold moment for the wrong secondary structure (FP/(FP+TN)). Each point in the plot corresponds to a particular value of the decision threshold. The $y = x$ line is what would be achieved by randomly guessing "yes." More conservative rules, those with lower thresholds and which say "yes" less often, occur toward the lower left of the curves. Both alpha and beta moment decision rules are better than random, and the alpha rule appears to be more discriminating than the beta rule.

7.4 Method

In order to find out whether certain types of information increase the secondary structure prediction accuracy of a neural network, I set up several experiments. In

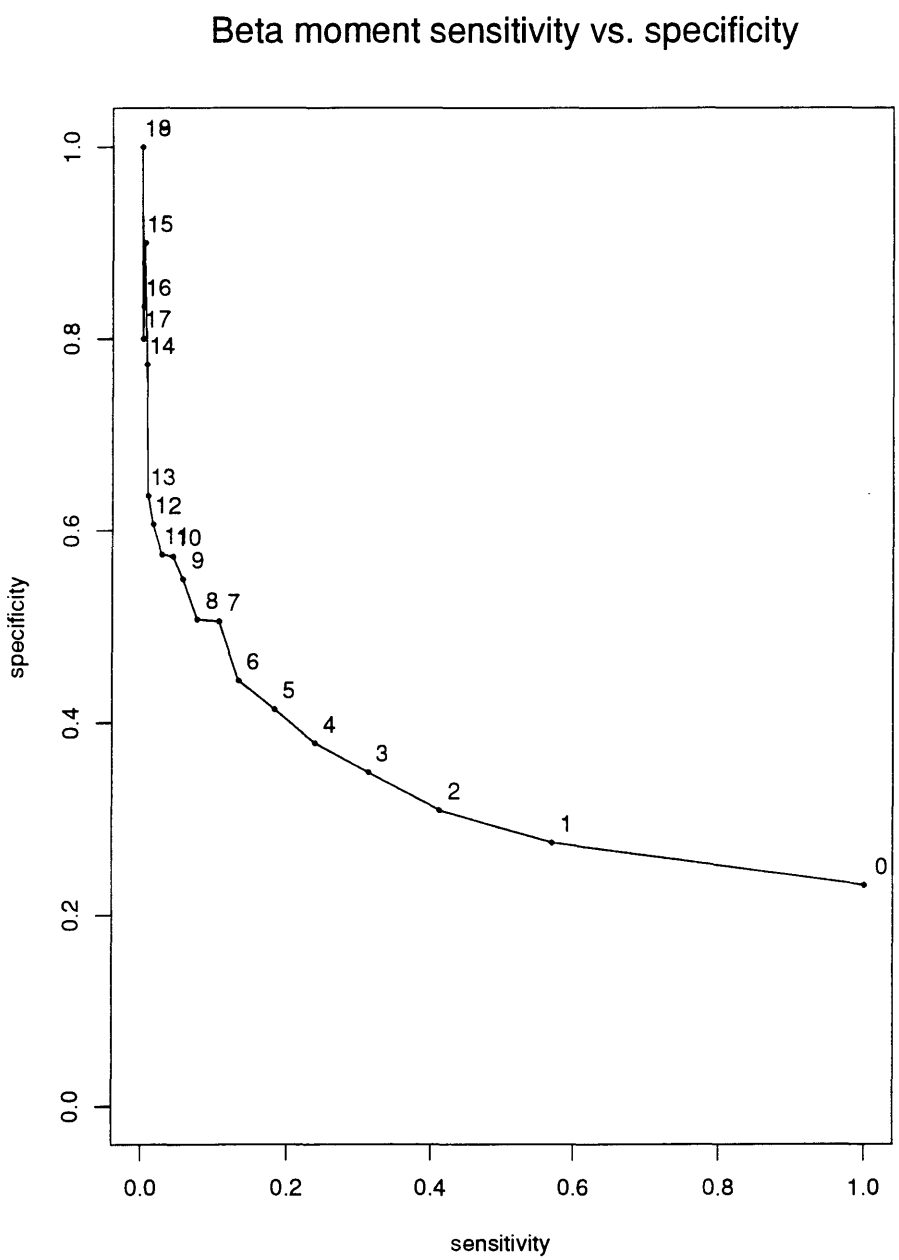


Figure 7-3: Beta moment sensitivity vs. specificity

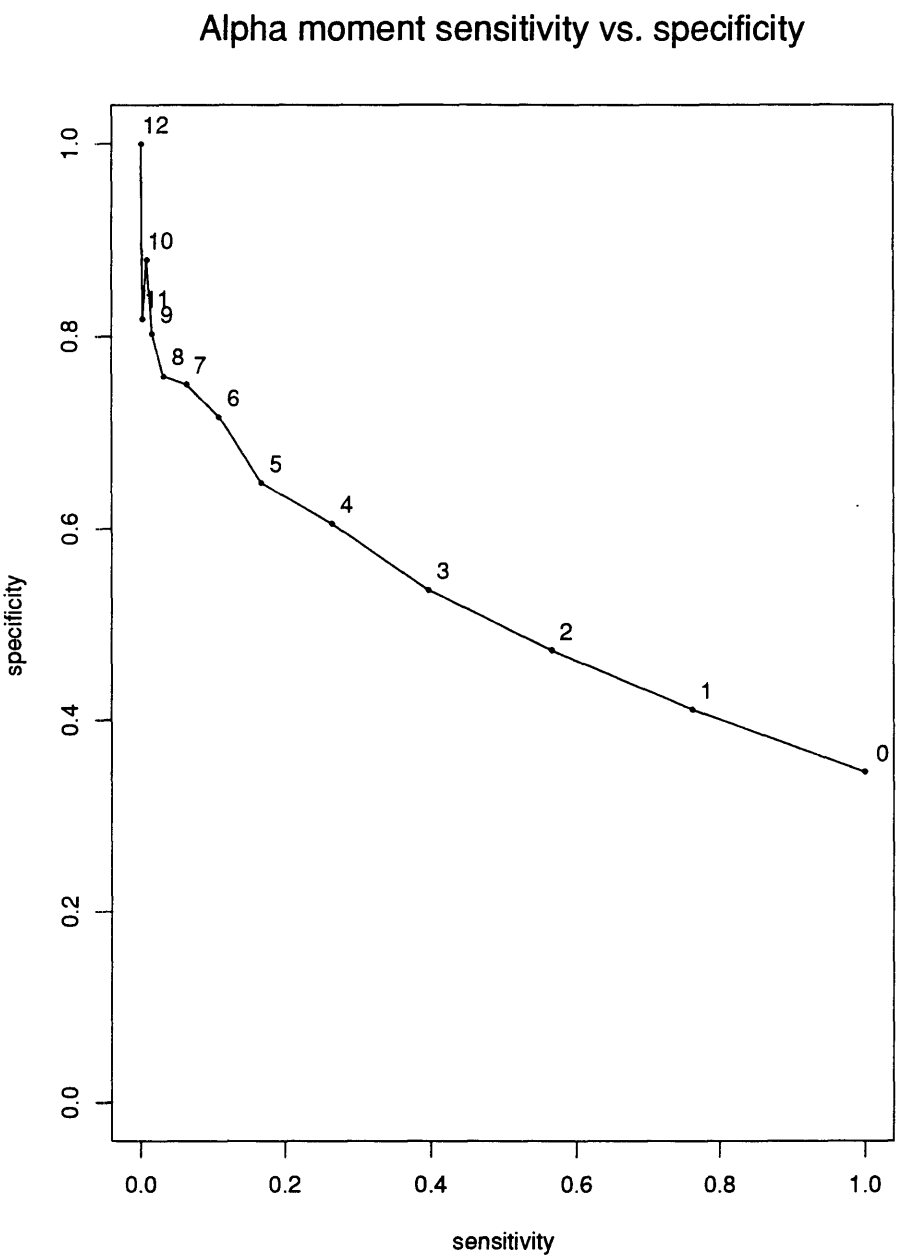


Figure 7-4: Alpha moment sensitivity vs. specificity

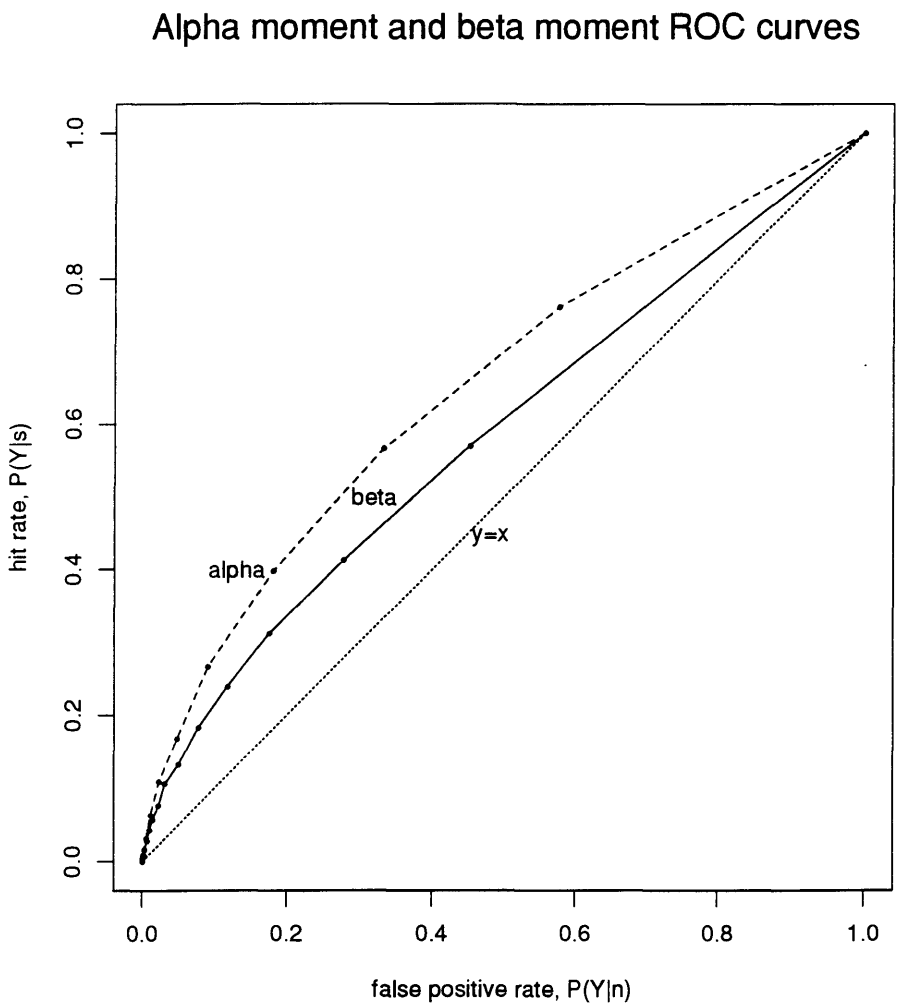


Figure 7-5: Alpha and beta moment receiver-operating curves.

each experiment the information provided to the network is different. I then can compare the performance of two experiments to determine whether added information improves performance.

7.4.1 Neural network

A neural network can be thought of as a paramatrized nonlinear function that maps an input vector to an output vector. A network is "trained" by adjusting the parameters, or "weights," to minimize the difference between the observed and desired outputs. In this chapter I use neural networks to determine roughly how much information relevant to structure is available in various representations of protein sequence.

I used neural networks with a single layer of weights (no hidden units). The network computes the following function of its input vector x:

$$y = f(Wx),$$

where x is an input vector of length M, y is an output vector of length N, W is an N by M weight matrix, and f is a nonlinear "squashing function" that operates on each member of its input vector, indexed by j, as

$$f_j(u_j) = \frac{1}{1 + e^{-u_j}}.$$

The squashing function is sketched in Figure 7-6. It has the property that it takes numbers from the range $[-\infty, +\infty]$ and outputs numbers between 0 and 1.

When a neural network is used as a classification method, the predicted class is taken to be the class corresponding to the output unit that has maximum value:

$$C = \max_j \{y_j\}.$$

Adding weight layers to the neural network gives the network more representational power. Networks with more layers are in principle capable of computing arbitrarily complex continuous functions, such as parity and amphipathicity from sequence.

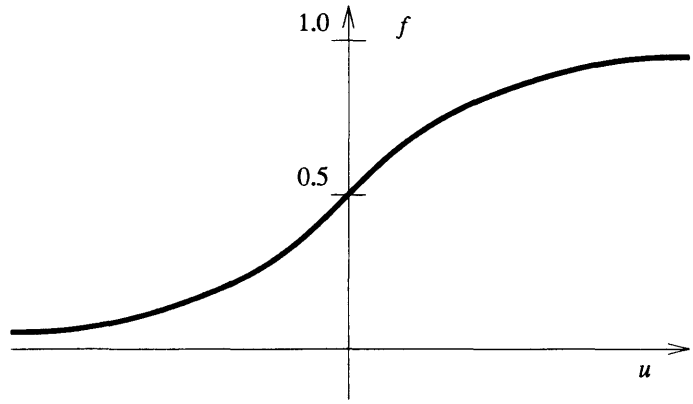


Figure 7-6: Sketch of the neural network node function.

The neural network training algorithm adjusts the weight matrix W to minimize the error over a set of n training examples (x_k, d_k), where x_k is the input vector and d_k is the desired or target output vector. For classification problems, each output node represents one class. The target output vector then has a 1 at the node corresponding to the correct class and 0 at each other node. The error measures the difference between the desired and predicted outputs, and is defined as

$$E = \sum_{k=1}^{n} (y_k - d_k)^2 ,$$

where $y_k = f(Wx_k)$ is the output of the neural network.

To minimize the error, the derivative of E is computed with respect to each weight w_{ji} in the weight matrix W. The weights are adjusted to reduce E:

$$w_{ji}(t) = w_{ji}(t-1) - \eta \left(\delta E / \delta w_{ji}\right) ,$$

where t indexes the iteration, and η is the learning rate. In addition, a "momentum" term is used that remembers the value of the previous weight update and adds a term proportional to it each time. The weight update is performed after each presentation of a training example to the network. The weights are initialized to small random numbers.

I used the Aspirin/MIGRAINES software, Release V6.0, for neural network train-

ing [Leighton, 1993].

The backpropagation training algorithm was used, with learning rate 0.001 and inertia 0.2. The nets were trained for 2,000,000 time steps, where a time step is one presentation of an example followed by a weight update. The weights were recorded every 100,000 time steps to allow determination of whether the nets were still learning.

As each neural network was built, the weights were saved before training began, and at 20 different time points during training.

The order of presentation of data to the network can affect the training. The training data were interspersed within cross-validation groups so that no two examples presented to the network in a row came from the same protein.

7.4.2 Data

I used the 130 proteins of known structure chosen by Rost and Sander [Rost and Sander, 1993b] for their neural network prediction work, and divided them into ten sets (Table 7.4) for cross-validation. Each experiment was run ten times, one for each cross-validation set. For each run, the 13 proteins in the cross-validation set were held out for testing, and the remaining 117 proteins were used as the training set.

The total number of examples was distributed among the secondary structure types as follows:

coil	11424	47%
alpha	7975	32%
beta	5114	21%
total	24513	100%

7.4.3 Output representation

I trained neural networks to predict three categories of secondary structure: helix, beta strand, and coil, as defined by Kabsch and Sander in their DSSP program. Residues Kabsch and Sander labeled H (α-helix), G (3,10-helix), or I (π-helix) were

Group	Proteins	Alpha	Beta	Coil	Total
1	256ba 3rnt 1lap 1etu 1bbpa 3tima 1paz	962	622	1337	2921
	2glsa 3cla 1prcc 4rhv1 1il8a 6dfr	33%	21%	46%	
2	9apib 2stv 2or1l 2gbp 2ccya 1wsya 1ppt	747	366	823	1936
	2i1b 2cyp 1ak3a 1sh1 5lyz 1fxia	39%	19%	43%	
3	7cata 2utga 1r092 3hmgb 1crn 1acx 1s01	879	422	1208	2509
	1lrd3 1fkf 3blm 4ts1a 3pgm 1hip	35%	17%	48%	
4	6cpa 2aat 7rsa 5ldh 1fc2c 1bds 6tmne	805	393	1330	2528
	9pap 2gn5 3cln 1prch 4rhv3 9insb	32%	16%	53%	
5	3ebx 1azu 2tgpi 1ovoa 2gcr 1cd4 1wsyb	381	599	962	1942
	1rbp 3icb 5cytr 2alp 2sns 1mcpl	20%	31%	50%	
6	4fxn 1cbh 9wgaa 2mhu 2hmza 1csei 8adh	792	502	1026	2320
	3sdha 2ltna 2fnr 4bp2 2tsca 2phh	34%	22%	44%	
7	6hir 6cpp 8abp 2rspa 2lh4 1fdlh 1bmv1	864	571	1064	2499
	2tmvp 2pcy 1gp1a 4cms 1prcl 4rhv4	35%	23%	43%	
8	1l58 5er2e 3b5c 1tgsi 2paba 1gd1o 1cdta	947	686	1236	2869
	4xiaa 1rhd 7icd 1eca 9apia 2sodo	33%	24%	43%	
9	2mev4 3gapa 1cc5 2wrpr 1mrt 5hvpa 6cts	539	278	780	1597
	3ait 4sgbi 2ltnb 2fxb 2cab 1ubq	34%	17%	49%	
10	1pyp 3hmga 4cpv 6acn 4rxn 2lhb 1fdx	1059	675	1658	3392
	1bmv2 1tnfa 4pfk 4gr1 4cpai 1prcm	31%	20%	49%	

Table 7.4: Proteins used in the neural network secondary structure experiment, grouped by cross-validation set. Each protein is referred to by its four-character Brookhaven Protein Data Bank name. If the protein is one subunit in the data file, the subunit character is appended to the protein name. For example, 1tgsi is the protein 1TGS, subunit I. For each group, the number and percent composition of alpha, beta, and coil residues is shown.

defined to be helix ("alpha"); residues labeled E were defined to be strand ("beta"), and all other residues were defined as "coil."

The output representation was a 3-vector, one output node corresponding to each secondary structure category. The secondary structure representation used was

alpha	[1.0, 0.0, 0.0]
beta	[0.0, 1.0, 0.0]
coil	[0.0, 0.0, 1.0]

7.4.4 Input representation

The number of inputs to the networks depended on the input representation used. The network inputs represented a 13-residue window on the sequence.

Amino acid encoding

Three different input representations were used for amino acid residues: the bit representation, secondary structure propensities, and hydrophobicity. Hydrophobicity was used both alone and with the secondary structure propensities.

- The bit representation of an amino acid is a 21-vector. All the components are 0 except for the one corresponding to the amino acid. The 21st element corresponds to the situation where the sequence window is off the edge of the protein.
- The secondary structure propensities are those defined by Levitt [Levitt, 1978] (Pa for alpha helix, Pb for beta strand, and Pt for turn).
- For hydrophobicity, I used Ponnuswamy's hydrophobicity index [Ponnuswamy *et al.*, 1980].

As a control, four random properties were used corresponding to the secondary structure propensities and hydrophobicity. Each random property was generated using a uniform probability distribution over the interval spanned by the corresponding residue property. For example, the random "Pa" property was generated with a

uniform distribution on $[0.52, 1.47]$; 0.52 is the minimum Pa value and 1.47 is the maximum Pa value.

The residue property scales (secondary structure propensities, hydrophobicity index, and random scales) are shown in Table 7.5.

Hydrophobic moments

In addition to the amino acid encoding, two other units were used to encode I_α and I_β, as described in Section 7.3.1 and Figure 7-1).

Input representations for the ten experiments

The input encodings for the ten experiments are shown in table 7.6. The total number of inputs and the total number of weights is also shown for each experiment. There is one weight connecting each input with each output node, and there are three additional nodes connecting the always-1 offset or bias node to the output nodes.

7.4.5 Network performance

Percent correct

To measure the performance of a trained neural network, I computed the fraction of predictions which were correct, both overall and for each of the secondary structure types individually. This number is reported as a percentage.

Cross-correlation coefficient

I computed the cross-correlation coefficient,

$$CC_i = \frac{TP_i TN_i - FP_i FN_i}{\sqrt{(TP_i + FP_i)(TP_i + FN_i)(TN_i + FP_i)(TN_i + FN_i)}},$$

where i represents the type of secondary structure; TP is the number of true positives (correct predictions of that secondary structure); TN is the number of true negatives (correctly predicting a different structure); FP is the number of false positives (incor-

	Pa	Pb	Pt	H	RPa	RPb	RPt	RH
I	0.97	1.45	0.51	1.81	1.05	1.18	1.36	-0.48
W	0.99	1.14	0.75	1.71	0.82	1.48	1.61	-0.06
F	1.07	1.32	0.58	1.35	0.71	1.06	1.59	0.97
L	1.30	1.02	0.59	1.14	1.29	0.71	0.97	-0.73
V	0.91	1.49	0.47	1.13	1.38	0.68	0.85	1.18
Y	0.72	1.25	1.05	1.11	1.20	1.42	0.72	0.54
M	1.47	0.97	0.39	1.00	0.75	1.04	1.08	-0.80
C	1.11	0.74	0.80	0.77	1.34	0.66	0.75	1.46
H	1.22	1.08	0.69	0.26	1.37	1.37	0.66	0.11
A	1.29	0.90	0.78	0.02	1.40	1.25	0.95	-0.17
-	0.50	0.50	1.00	0.00	0.54	1.07	1.10	-0.49
X	1.00	1.00	1.00	0.00	1.33	1.08	1.47	0.20
P	0.52	0.64	1.91	-0.09	0.61	1.00	0.72	-0.01
K	1.23	0.77	0.96	-0.41	1.34	1.07	1.15	1.36
R	0.96	0.99	0.88	-0.42	1.36	0.93	1.38	-0.89
N	0.90	0.76	1.28	-0.77	0.97	1.09	0.69	-0.02
T	0.82	1.21	1.03	-0.77	1.36	1.14	1.43	0.05
G	0.56	0.92	1.64	-0.80	1.19	0.75	1.21	1.34
B	0.97	0.74	1.35	-0.91	0.96	0.94	0.60	-0.42
S	0.82	0.95	1.33	-0.97	1.37	1.05	0.83	0.57
D	1.04	0.72	1.41	-1.04	0.95	0.80	0.50	-0.82
Q	1.27	0.80	0.97	-1.10	0.65	1.09	0.83	-0.95
Z	1.35	0.77	0.99	-1.12	0.83	0.78	0.82	-0.78
E	1.44	0.75	1.00	-1.14	0.56	1.28	1.56	0.25

Table 7.5: Residue encoding for the neural network. Pa, Pb, and Pt are from Levitt; the hydrophobicity scale is that of Ponnuswamy (H). In addition, the random scales are shown: RPa – random "Pa"; RPb – random "Pb"; RPt – random "Pt"; RH – random "hydrophobocity". X: any residue. B: Asn (N) or Asp (D). Z: Gln (Q) or Glu (E).

Input representation		Experiment									
Residue encoding	# inputs	1	2	3	4	5	6	7	8	9	10
		B	BA	P	PA	H	HA	RP	RPA	RH	RHA
Bit	273	X	X								
Pa, Pb, Pt	39			X	X						
Hydrophobicity	13			X	X	X	X				
α and β moments	2		X		X		X				
Rand. "Pa, Pb, Pt"	39							X	X		
Rand. "hydrophobicity"	13							X	X	X	X
Rand. α, β moments	2								X		X
Total inputs		273	275	52	54	13	15	52	54	13	15
Total weights		822	828	159	165	42	48	159	165	42	48

Table 7.6: Neural network experiments, showing the input representation for each experiment. Each row corresponds to an element of the representation. Columns corresponding to each experiment are marked with an X at the representation elements used for that experiment.

rectly predicting that structure); and FN is the number of false negatives (incorrectly predicting a different structure). Values range between -1 (completely wrong) and 1 (complete right), with 0 corresponding to the performance that would be obtained by guessing at random. The cross-correlation coefficient gives a good indication of the prediction performance for each of the type of secondary structure. For example, the fraction of correct predictions for coil reports only the ratio of true positives and true negatives ($TP+TN$) to the total number of predictions. If coil is overpredicted, then this number can be quite high. On the other hand, the correlation coefficient also takes into account the false positives and false negatives, and therefore is more informative.

ROC curves

ROC curves were computed for alpha, beta and coil structure as described in section 7.3.3. These summarize the predicted vs. correct tables of counts for a decision rule made by thresholding the corresponding output unit of the neural network. Varying the threshold generates the curve of hit rate vs. false positive rate.

7.4.6 Significance tests

I used statistical tests to determine the signficance of observed increases in prediction performance, and to ask whether the neural networks had finished training.

Improved prediction performance

My analysis of the neural network results involved comparing two of the experiments and asking whether one showed significantly better prediction than the other. If so, I concluded that the input representation for the experiment that performed better contained more information relevant to the prediction problem.

I used a t test to determine the significance of the difference in prediction performance of two experiments. For each cross-validation group in each experiment, I computed the average prediction performance on the train and test sets over the last five time points at which the network weights were recorded. The difference in this prediction performance is d_i for the i'th cross-validation group. Call the average difference across all cross-validation groups $\overline{d}$. I examined the null hypothesis that the distribution of the d_i values is centered at 0. This would be interpreted as meaning there is no difference in prediction performance between the experiments. Assuming that the d_i values are normally distributed, we can use the t distribution to check the null hypothesis.

The reference distribution against which the observed $\overline{d}$ may be viewed is a scaled t distribution centered at 0. Because there are $n = 10$ cross-validation groups, or d_i values, the t distribution has $n - 1 = 9$ degrees of freedom. The distribution has a scale factor of $s_d/\sqrt{n}$, where s_d is the standard deviation of d_i. The value of t_0 associated with the null hypothesis is

$$t_0 = \frac{\overline{d}}{s_d/\sqrt{n}},$$

which can be referred to a t table with n degrees of freedom.

Training

To test whether the training of a network has completed, I look at the training performance within each cross-validation group. I compute the average training performace over time points 11 through 15 and time points 16 through 20. I then compare these averages to determine the difference in performance, and decide that learning is completed if the distribution of differences has a mean not significantly different from 0.

7.5 Results

In this section, I describe the prediction performance of the trained networks, the changes in performance with the various input representations, and the issue of completion of learning.

7.5.1 Performance

Appendix C shows the predicted vs. correct counts for each experiment. Table 7.7 summarizes the accuracy of each neural network after training.

The train and test results are quite similar, indicating that there are sufficient data in the training set to adequately represent the information necessary for performing prediction on the test set. The largest discrepencies between the train and test sets occur for the experiments with the most free parameters (weights) in the neural network. This is not surprising, as the larger number of free parameters allows a greater "memorization" of the training data. On the other hand, an increased number of parameters is also correlated with better results, suggesting that the larger input representations contain more information.

7.5.2 Amphipathicity

Adding the hydrophobicity information improves the results by a small but significant amount. Table 7.8 shows the change in average results when two inputs units

Experiment		Train		Test							
		ave.	s.d.	ave.	s.d.	α	β	coil	CC_α	CC_β	CC_c
1	B	62.62	0.27	61.81	2.04	58.77	35.08	76.29	38.87	33.41	40.98
2	BA	64.56	0.57	62.41	2.51	61.19	35.75	75.84	41.88	33.86	41.30
3	P	59.49	1.20	59.16	2.81	52.22	32.78	76.49	33.45	30.96	37.42
4	PA	61.05	1.08	60.21	3.22	56.55	34.88	74.98	37.09	32.41	38.43
5	H	51.31	0.27	51.20	2.23	21.20	25.67	84.07	9.34	22.97	24.29
6	HA	54.55	0.59	54.65	2.07	40.81	29.50	76.12	23.19	25.59	29.04
7	RP	48.23	2.11	47.60	4.81	35.00	6.08	74.82	13.68	4.83	12.22
8	RPA	48.66	1.53	48.31	3.90	40.17	3.74	73.09	15.52	4.69	13.34
9	RH	48.00	0.34	48.01	3.26	23.05	0.00	87.18	14.74	0.00	7.27
10	RHA	47.90	0.42	47.66	3.13	22.81	0.02	86.20	13.41	-0.03	6.63

Table 7.7: Summary of neural network results. Numbers in table are percent correct predictions, at the end of 2×10^6 training iterations. Names of experiments are as follows: B – bit encoding; P – secondary structure propensities and hydrophobicity; H – hydrophobicity alone; A – alpha and beta hydrophobic moments; R – random. For the training and test sets, the average and standard deviation of performance are given in percentage points, averaged across all ten cross-validation groups. In addition, average alpha, beta and coil predictions are presented in percentage points for the test set, as well as the average test set cross-correlation coefficients.

representing the alpha and beta hydrophobic moments are added to the network. For nonrandom inputs (experiments 1–6), there is a significant improvement in performance ($P < 0.01$). Table 7.8 also shows the average difference across cross-validation groups, the standard deviation, and the value of t_0. The random input representation actually shows a significant decrease in performance in the training set with the addition of the two amphipathicity nodes. This may be due to the destabilizing effect of the hydrophobic moment inputs on the random-input networks (see Section 7.5.4 on learning curves).

The improvement in performance is smallest for the best-performing input representation.

The importance of doing the significance test on the performance improvement within cross-validation groups can be seen. Looking at the difference in averages is not sufficient, given the large standard deviations of the test set performances.

Experiments		$\bar{d}$	s_d	t_0	P
1,2	train	0.84	0.22	12.38	<0.001
	test	0.83	0.68	3.85	0.004
3,4	train	1.64	0.53	9.88	<0.001
	test	1.36	0.83	5.14	<0.001
5,6	train	3.12	0.35	28.32	<0.001
	test	3.21	1.80	5.64	<0.001
7,8	train	0.37	1.66	0.70	0.50
	test	0.58	2.50	0.74	0.48
9,10	train	-0.38	0.20	-6.04	<0.001
	test	-0.25	0.87	-0.90	0.39

Table 7.8: Improvement in results with I_α and I_β.

Experiments		$\bar{d}$	s_d	t_0	P
9,5	train	3.31	0.25	41.79	<0.001
	test	3.23	2.12	4.81	0.001
5,3	train	8.17	0.45	57.15	<0.001
	test	8.02	2.95	8.61	<0.001
3,1	train	4.13	0.38	34.45	<0.001
	test	2.66	1.49	5.63	<0.001
4,2	train	3.33	0.67	15.75	<0.001
	test	2.13	1.48	4.55	0.001

Table 7.9: Comparison of other experiment pairs

7.5.3 Amino acid encoding

Table 7.9 shows comparisons between several other pairs of experiments.

Representing amino acids by their hydrophobicity alone is a clear improvement over a random scale (experiments 9 and 5). Adding secondary structure propensities to the hydrophobicity improves the results significantly (experiments 5 and 3). The number of input nodes representing each amino acid goes from one to four when the secondary structure propensities are added.

The 21-bit residue representation, which is the one most often used in work by other researchers, performs better than the four-input propensity representation. The difference in performance seen in going from 4-input to 21-input representation (for a

total of 221 more inputs across the 13-residue input window) can be examined both for the experiment pair (3,1) (no hydrophobic moments) and for the experiment pair (4,2) (with hydrophobic moments). These numbers are 4.13%/2.66% ($P < 0.001$) for experiments (3,1) and 3.33%/2.13% ($P \leq 0.001$) for experiments (4,2).

7.5.4 Learning Curves

The network weights were saved every 100,000 time steps during training to ensure that the network had finished training.

Figure 7-7 shows the average (over cross-validation groups) prediction performance as a function time for each experiment, on the training (dashed) and test (solid) sets.

Appendix C shows the learning curves over time for all cross-validation groups in the 10 experiments. Interestingly, adding the two hydrophobic moment inputs seems to create more erratic learning curves in some of the networks (for example, compare the curves for experiments 1 and 2). This effect would be reduced by lowering the learning rate η. Also of interest is that adding three random scales to a single random scale results in much more erratic learning (compare curves for experiments 7 and 9).

I compared the performance average over time steps 11 through 15 with that over time steps 16 through 20 to determine whether learning was complete. The results are shown in Table 7.10. Only those networks that used the bit input representation showed significant positive change over these time periods. As the bit representation networks are the best performers, and my results would not be changed by an improvement in their performance, I did not continue the training.

7.5.5 ROC curves

ROC curves were drawn for a couple of representative networks. The best-performing experiment, experiment 2, is represented in Figure 7-8 by its first cross-validation neural network.

For contrast, I show the ROC curves for the random-representation neural network of experiment 8 in Figure 7-9.

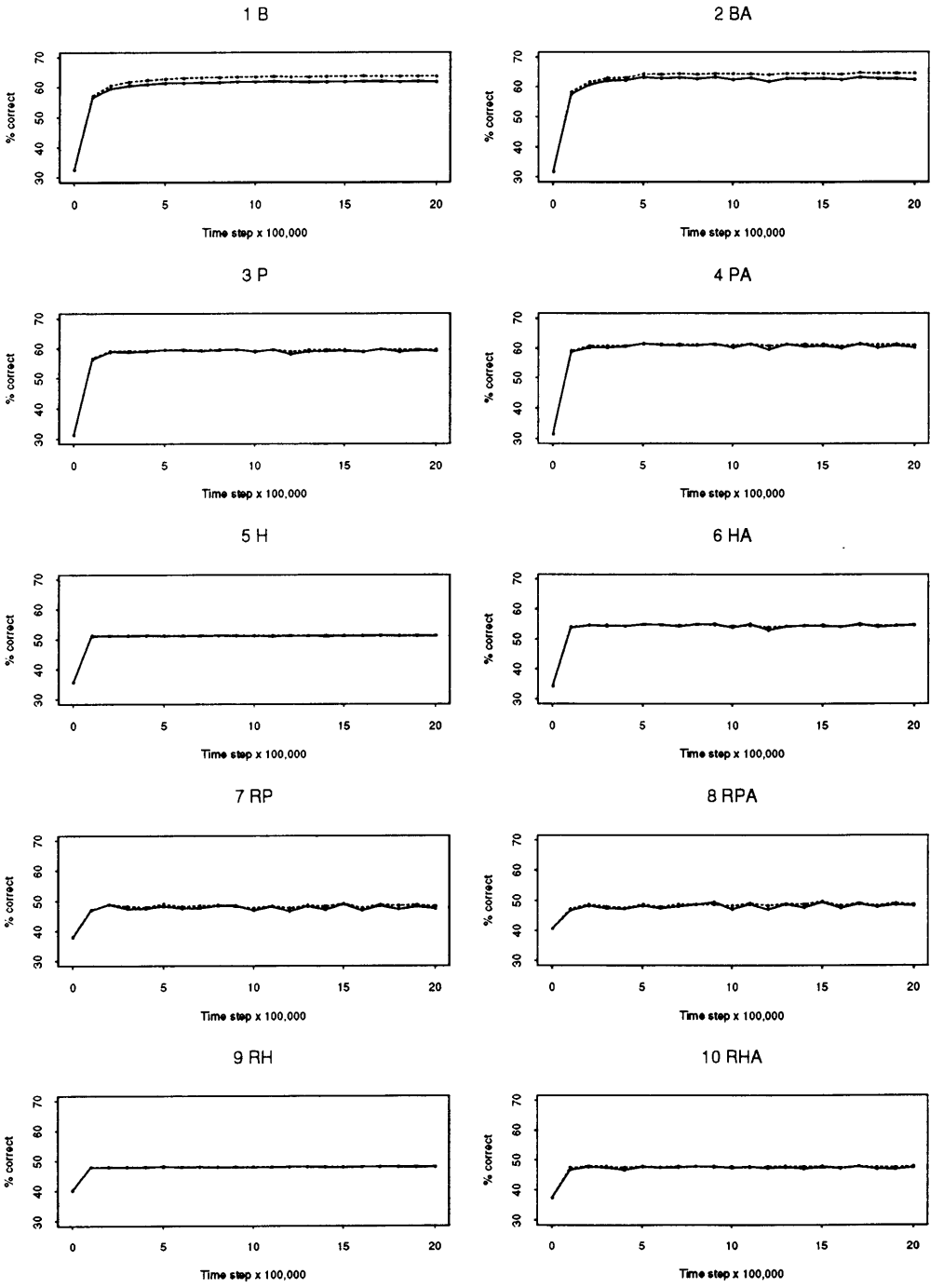


Figure 7-7: Neural net performance during training. Dashed lines are performance on training set; solid lines are performance on test set. Each curve is averaged over the ten cross-validation groups.

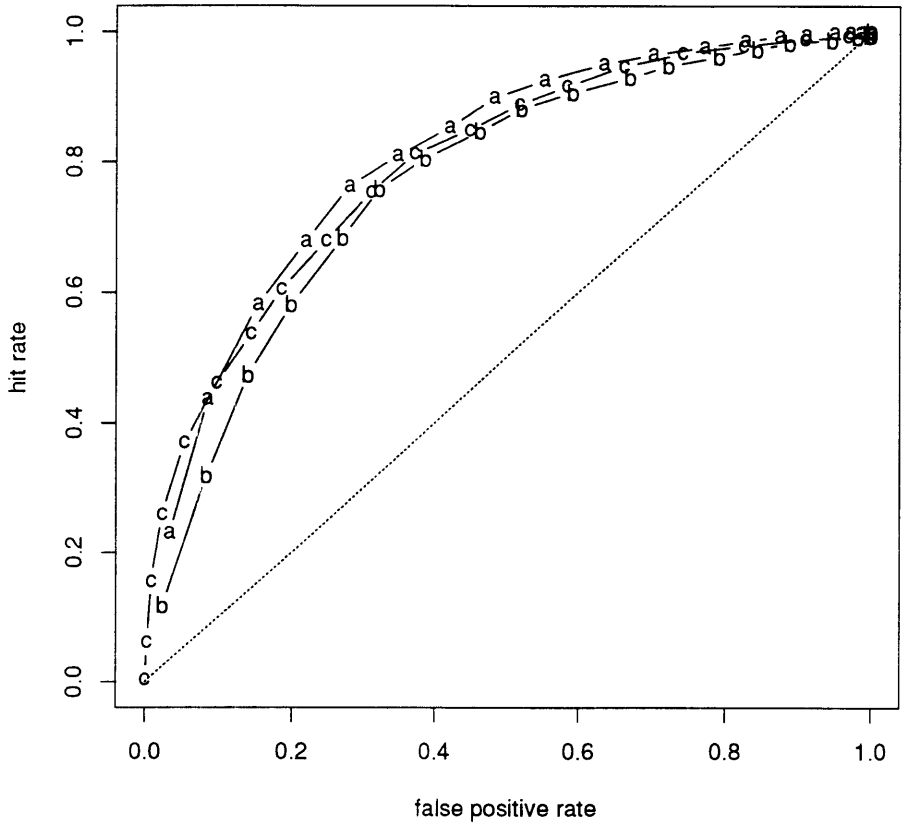


Figure 7-8: ROC curves from experiment 2, cross-validation group 1. The curves are labeled "a" for alpha, "b" for beta and "c" for coil.

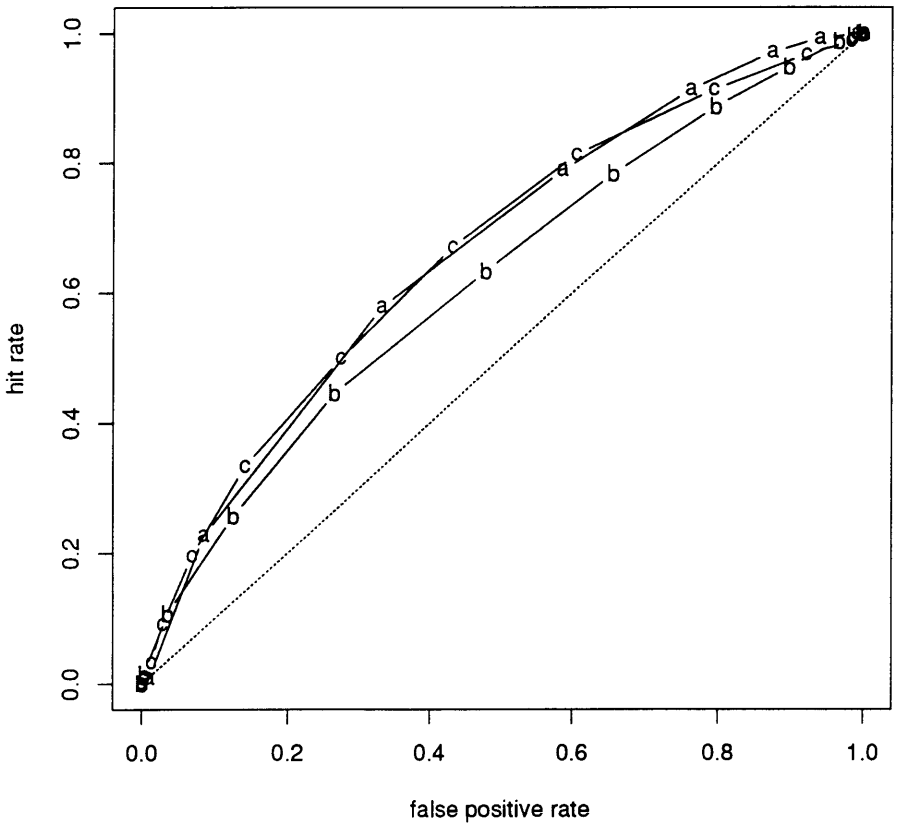


Figure 7-9: ROC curves from experiment 2, cross-validation group 1. The curves are labeled "a" for alpha, "b" for beta and "c" for coil.

Experiment		$\bar{d}$	s_d	t_0	P
1	B	0.073	0.080	2.89	0.02
2	BA	0.184	0.363	1.60	0.14
3	P	0.040	0.632	0.20	0.84
4	PA	-0.071	0.623	-0.36	0.73
5	H	-0.004	0.052	-0.27	0.80
6	HA	0.167	0.680	0.78	0.46
7	RP	-0.05	0.840	-0.21	0.84
8	RPA	-0.249	0.870	-0.90	0.39
9	RH	-0.001	0.031	-0.05	0.96
10	RHA	-0.120	0.162	-2.35	0.04

Table 7.10: Learning completion tests. $\bar{d}$ is the average difference (over the cross-validation training sets) of the mean training performance for time 11 through 15 and that of time 16 through 20. s_d is the corresponding standard deviation; P indicates the significance of the difference.

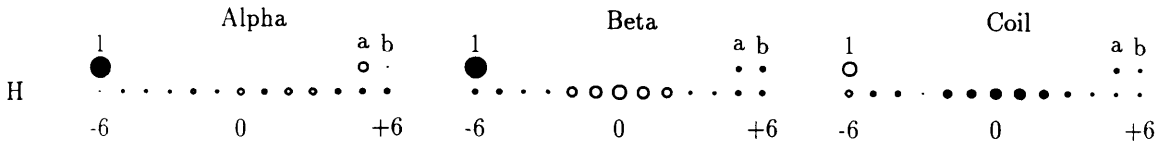


Figure 7-10: Hinton diagram for cv group 1 of experiment PO-PO. The largest weight magnitude is 1.53

7.5.6 Weights

I examined the weights from several of the networks using a variation of the diagrams developed by Hinton [Rumelhart *et al.*, 1986]. These are shown in Figures 7-10 through 7-14.

Weight diagrams

Each circle in a weight diagram represents the value of a single weight. The area of the circle is proportional to the magnitude of the weight. Black circles indicate weights less than zero and white circles indicate positive weights.

Each weight diagram has three parts corresponding to the three outputs alpha,

beta and coil. All the weights going from an input unit to the alpha output node are displayed in the left third of the diagram. The circle on the top left, labeled "1," represents the weight going from the constant unit, the input which is always 1, and which allows a constant offset to be added to each input vector.

If the neural network had inputs representing the alpha and beta moments, the weights from these inputs to the outputs are shown on the top right. The alpha moment weight (labeled "a") is shown to the left of the beta moment weight (labeled "b").

Each row below the constant unit circle corresponds to one element of the input representation for an amino acid. The precise meaning of the row depends on the input representation chosen. For the bit representation, each row corresponds to a particular amino acid (or the off-the-end indicator). Alternatively, the row might correspond to a secondary structure propensity, hydrophobicity, or a randomly-chosen scale.

Each column corresponds to one position in the 13-residue window. The leftmost column corresponds to the N-terminal end of the window, and the rightmost column corresponds to the C-terminal end.

Comparing the sizes of weights for different inputs can be misleading when the inputs have different ranges. For example, the beta moments range from 0 to over 20, whereas the secondary structure preferences range from 0.39 to 1.81. Multiplying the beta moment weight by the beta moment, then, usually results in a number of larger magnitude than most of the products of secondary structure preference and weight. The beta moment contribution is larger than might appear by a direct comparison of the circle sizes. The best way to use the circle sizes is to compare the sizes within rows, and between rows with similar ranges of input weights (such as any two bit representations, or any two secondary structure preferences). Hydrophobicity has a range which is about three times that of the secondary structure preferences.

Note as well, that the sizes of the circles are not calibrated from one figure to the next. The caption of each figure gives the magnitude of the largest weight.

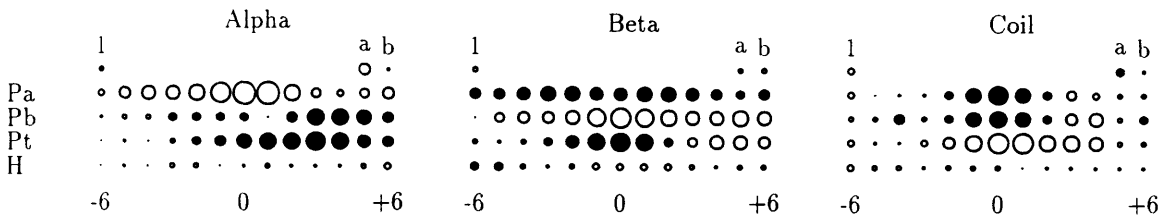


Figure 7-11: Hinton diagram for cv group 1 of experiment PO-PO-SS. The largest weight magnitude is 1.19

Constant offset weights

The constant offset weights reflect the overall secondary structure composition of the training set. The network is biased toward predicting coil (high, positive weight from the constant offset unit to the coil output), because about 47% of the training examples are coil, as opposed to 32% alpha and 21% beta.

Alpha and beta moment weights

The alpha and beta moment weights show that the moments contribute to the network in the way we would expect. For alpha secondary structures, the alpha moment is positively weighted and the beta moment is negatively weighted. The weights are reversed for beta secondary structure. And for coil, both moments have negative weights. The alpha and beta moment weights are not directly comparable because the range, and average value, of the beta moment is greater than that of the alpha moment.

Patterns of weights

The circles along a given row in the diagram, corresponding to the weights on one part of the input representation along the 13-residue window, generally take on one of a set of patterns. Often the magnitude of the weights is at a maximum near the center of the window, indicating that more of the information contributing to a secondary structure assignment comes from the residue itself, and less comes from neighbors

according to how distant they are.

Hydrophobicity

Figure 7-10 shows the weights for the experiment in which amino acids are represented only by their hydrophobicity index. In addition, the two hydrophobic moments are used as inputs. The alpha weights are small and varied, and show some asymmetry. The beta weight pattern is a central string of five hydrophobic residues surrounded by hydrophilic residues. The indices vary monotonically from strongly hydrophobic at the center to hydrophilic at the window edge. The coil weights are the reverse: there is a preference for hydrophilic residues at the center that diminishes toward the ends. There is a hydrophobic preference at the N-terminus.

The same patterns are observed in the weights from the hydrophobicity index inputs when the input representation is expanded to include property preferences, as shown in Figure 7-11. The alpha hydrophobicity weights in this network are asymmetric and include both positive and negative weights. The beta weights have the same pattern of hydrophobic residues preferred in the center, and hydrophilic at the outside of the window. The coil hydrophobicity weights prefer hydrophilic in the center, and hydrophobic toward the N-terminal end of the window.

Property preferences

The weights on links from the Pa, Pb, and Pt inputs to the output are shown in Figure 7-11. These show a clear favoring of Pa for alpha, Pb for beta, and Pt for coil.

Amino acid preferences

In Figures 7-12 and 7-13 we can see the weights for the bit representation.

For alpha secondary structure, the residues A, L, M, F, W and I have positive weights. The residues C, G, N, P, S, T, and V have negative weights. Moreover, the charged residues show an asymmetry. Negatively charged D and E have positive weights toward the N-terminus and negative weights toward the C-terminus. Positively charged H, K, Q, and R have the opposite pattern. This is a result of the

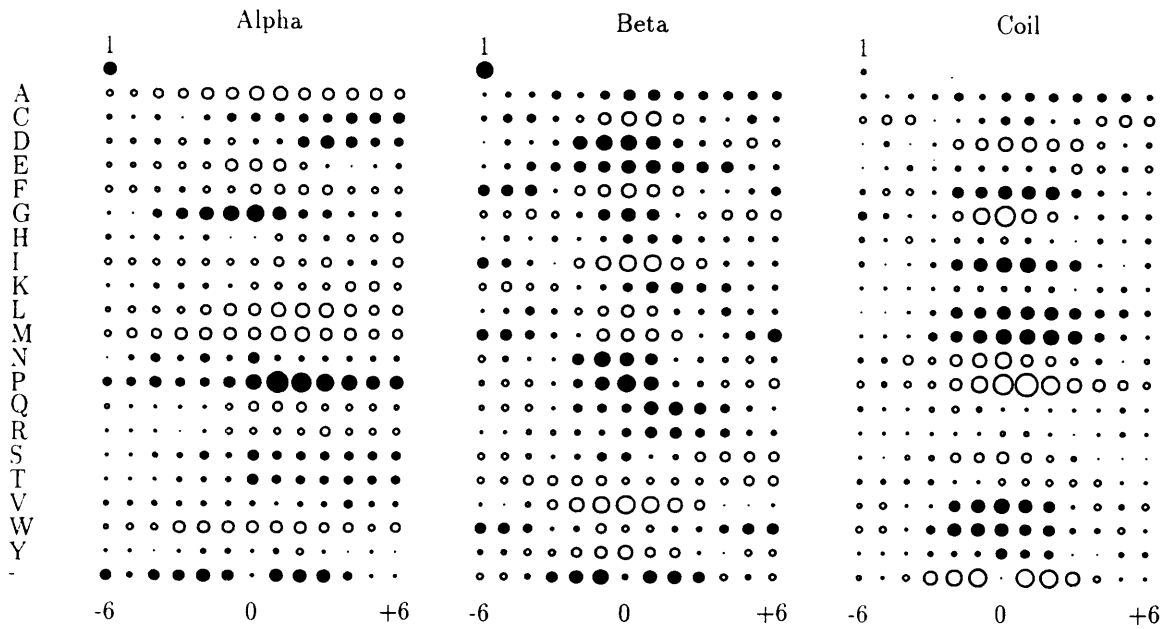


Figure 7-12: Hinton-like diagram for BI. The area of the largest circle represents the maximum weight absolute value of 1.97.

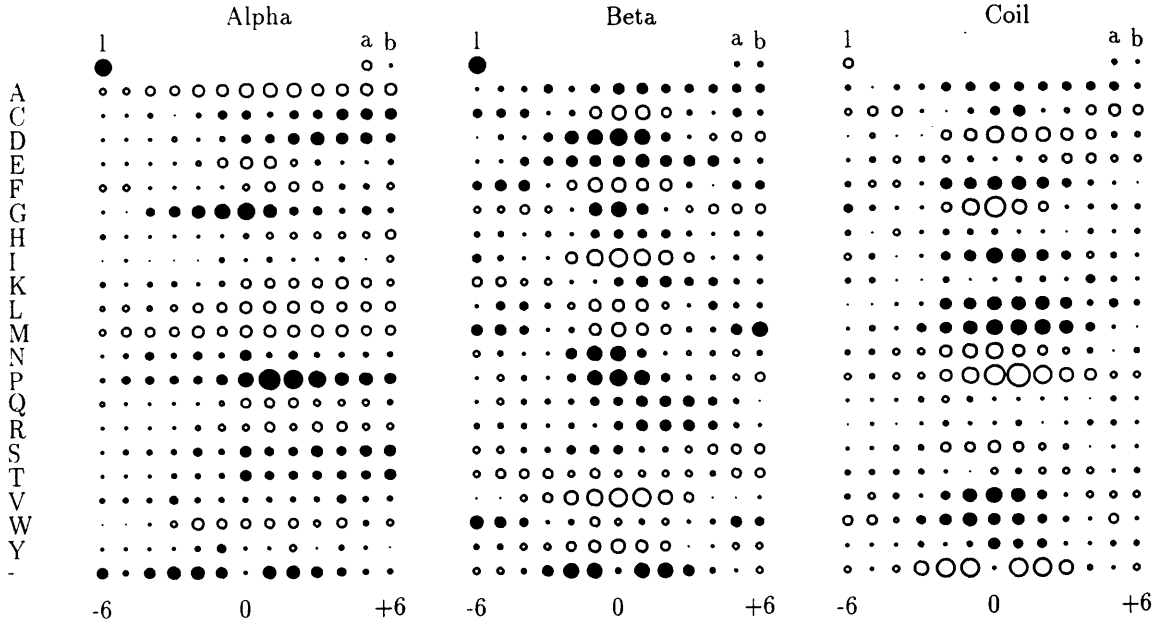


Figure 7-13: Hinton-like diagram for BI-PO. The largest weight magnitude is 1.92.

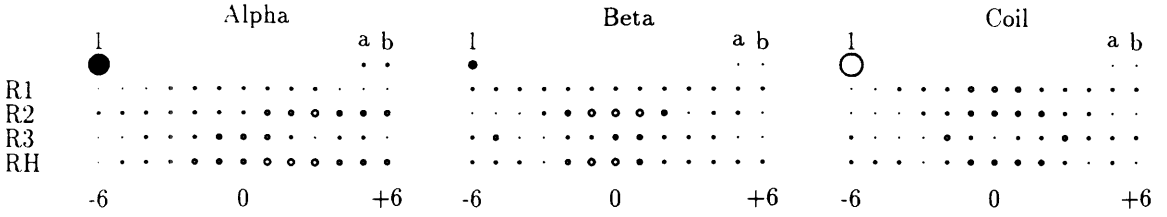


Figure 7-14: Hinton diagram for cv group 1 of experiment RA-RH-RS. The area of the largest circle represents the maximum absolute weight value of 6.37.

well-known tendency for negative residues to cap the positive N-terminal end of a helix, and for positive residues to cap the negative C-terminal end of a helix.

The beta weights show that T has positive weights across the window, while E and A have negative weights across the window. Other beta-favoring or -disfavoring residues have a weight pattern that is negative in the center and positive at the edges, or vice versa. This is due to the shorter length of beta strands relative to alpha helices. Residues which have positive weights in the center and negative on the edges include C, F, I, L, M, V and W. Residues which have negative weights in the center and positive on the edges include G, N, P, S, and the placeholder $-$. Interestingly, there are several charged residues which have an asymmetric pattern like that seen in the alpha weights, although in the beta weights the pattern is reversed. Here, the negatively charged D residue is preferred at the C-terminal end of the window, instead of the N-terminal end. And the positively charged residues K, Q, and R are preferred at the N-terminal end and disfavored at the C-terminal end. Why would this be? It is possible that the geometry of the strand provides for a strand capping mechanism that works differently than helix capping. Perhaps the preferences are a result of alternating alpha/beta structure, where the helix caps are picked up by the 13-residue beta window.

Random controls

The weights on a network trained on randomly-chosen structure preference and hydrophobicity scales are shown in Figure 7-14. The dominant weights are the constant offset weights. Compare these weights to the constant offset weights in Figure 7-11. The moment weights, on inputs computed from the random hydrophobicity scale, are very small. The other weights are also small. It appears that the random "Pb" scale has captured something related to hydrophobicity or the real Pb scale.

7.6 Conclusion

Previously, I showed that hydrophobicity patterns provide information about the protein structure. However, this observation does not directly imply that hydrophobicity patterns will improve a given protein structure prediction that takes advantage of various other sources of information, such as aligned sequence data or secondary structure propensities. In this chapter, I demonstrated that it is possible to use this information to improve secondary structure prediction.

Chapter 8

Threading

In this chapter I investigate aspects of the pseudopotential functions used in tertiary structure predictions performed with the threading algorithm. The threading method is a way of optimally (with respect to some pseudopotential, or score, function) aligning a protein sequence to a protein structure. Once this alignment has been performed. a structure model can be generated for the sequence based on the alignment.

There are several reasons to look at the components of threading score functions. One reason is to find ways of improving the score functions for structure prediction. What structure representation is best? What sequence representation should be used? The score functions I look at are computed from examples of known protein structures. How do we deal with problems of low sample size?

Another reason to examine the components of threading score functions is to examine which factors are most important in determining the protein fold. Is secondary structure or solvent exposure preference more important? The statistical analysis performed in the preceding chapters suggests that the solvent exposure preference of the different amino acid types is the dominant effect in folding; is this borne out in threading experiments?

The method that I use in this chapter is to test various scoring functions by performing self-threading in which a protein's own sequence is threaded onto its structure. The experiments confirm the statistical analysis showing that solvent exposure

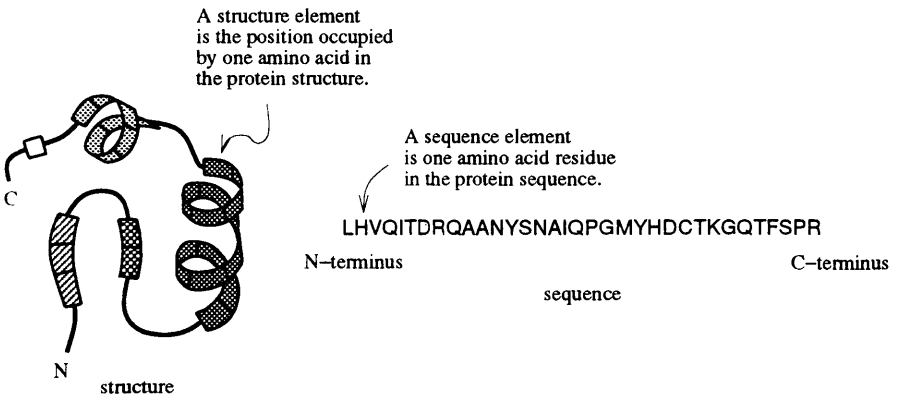


Figure 8-1: A protein structure and sequence. The elements of the protein structure are the positions occupied by amino acid residues. The elements of the protein sequence are the amino acid residue identities along the protein chain. The N-terminus and C-terminus are marked in the diagram; on the structure they are represented as "N" and "C", respectively.

is more predictive than secondary structure. I address the issue of subdividing structure categories and give solutions to problems of low sample size and lack of statistical significance.

8.1 Introduction

In this section I discuss the threading algorithm, the computation of threading scores, sample size problems and possible fixes, structure representations, and the incorporation of local sequence information.

8.1.1 Threading algorithm

The threading algorithm finds an optimal sequence-structure alignment relative to a score function. In this section I define alignments and explain how score functions are computed. I discuss the particular restrictions I make on the alignment between a protein sequence and a protein structure. Finally, I describe the dynamic programming algorithm for computing the optimal alignment.

Alignments

The goal in threading is to find a sequence/structure alignment to optimize a score or pseudopotential function. Figure 8-1 is a sketch of a protein structure and protein sequence. The elements of the sequence are the specific amino acids that compose it. For example, His occurs as the second sequence element in the figure. The elements of the structure are the positions in the structure which are occupied by amino acids. A structure element may be described by the three-dimensional coordinates of its backbone atoms. An alignment is a correspondence between the sequence and structure, such that each sequence element is matched to zero or one structure elements, and each structure element is matched to zero or one sequence elements. In addition, the alignments must preserve the linear order from N-terminus to C-terminus along the protein backbone of the sequence and structure. Figure 8-2 contains a sketch of a sequence-structure alignment.

We can define an alignment by a correspondence function C, such that $C(i,j) = 1$ if sequence element i and structure element j are aligned, and $C(i,j) = 0$ if sequence element i and structure element j are not aligned (see Figure 8-2(c)). As stated above, we require a maximum of one match per sequence and structure element: if $C(i,j) = 1$, then $C(i,j') = 0$ for all $j' \neq j$, and $C(i',j) = 0$ for all $i' \neq i$. Moreover, we require that order be preserved: if $C(i_1,j_1) = 1$, $C(i_2,j_2) = 1$, and $i_2 > i_1$, then $j_2 > j_1$.

Score functions

There are many types of score functions, and a given score function may be composed of multiple components. A "singleton" score term gives a score for the alignment of a sequence element and a structure element, and represents the preference of that amino acid for that structure type. A "pairwise" term gives a score for the simultaneous placement of two amino acids in a specified pair of structure positions. The structure positions in the pair are chosen because they are close enough together that the amino acid residues filling those positions would interact with each other. This is a way of modeling interactions between positions far apart in sequence, but close together in the protein structure. For a pairwise score function, the structure model

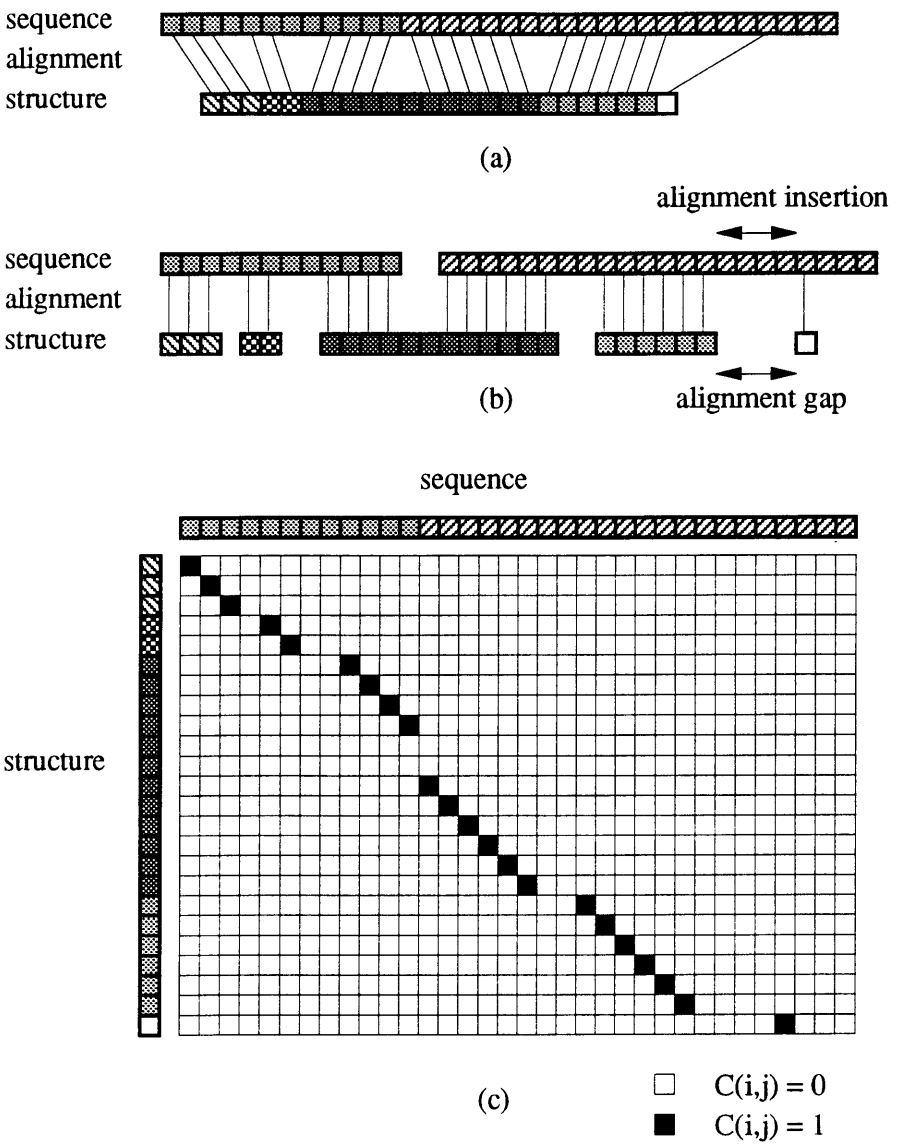


Figure 8-2: Sketch of an alignment. (a) A sequence is aligned to a structure. A linear set of boxes represents elements of the sequence and structure. The alignment is shown as lines connecting pairs of aligned elements. (b) Same as (a), except the sequence and structure are broken so that aligned elements occur in a vertical relationship to each other. Alignment gaps and insertions are clearer to see in this representation. (c) A representation of the alignment correspondence function C. Filled boxes ($C(i,j) = 1$) indicate that elements i and j are aligned.

must include a list of all structure position pairs to be considered. It is also possible to have tertiary and even higher-order terms in the score function. I will discuss the threading algorithm in terms of singleton scores only, to make the presentation easier to follow.

The score function has two parts, one that defines the value of a match between a sequence element and a structure element, and the other that defines the score for a gap or insertion in the sequence or structure. An alignment insertion in the sequence corresponds to an alignment gap in the structure, and vice versa. A sequence alignment insertion is a set of adjacent sequence elements which are not matched to any elements in the structure.

The total score for an alignment is the sum over all aligned element pairs of the match score, plus the insertion score over all insertions in either sequence or structure. I use an insertion score that is a straight sum of individual insertion scores for each element. Define the match score between sequence element i and structure element j as $M(i,j)$, the score for inserting unmatched sequence element i as $I_A(i)$, and the score for inserting unmatched structure element j as $I_S(j)$. Then the total alignment score is

$$S(C) = \sum_{i,j} C(i,j)M(i,j) + \sum_{i}(1 - Q_A(i))I_A(i) + \sum_{j}(1 - Q_S(j))I_S(i).$$

$Q_A(i)$ is an indicator function telling whether sequence element i has been aligned to any structure element (1 means yes and 0 means no), and $Q_S(j)$ is the corresponding indicator function for structure element j. Q_A can be computed as $Q_A(i) = \sum_j C(i,j)$; Q_S similarly.

Structure models

In this chapter I will use a specialized structure model in which I assume that the helices and strands in the structure should be aligned to the sequence with no gaps or insertions occurring internally to the secondary structure object. In addition, I allow sequence insertions of any length to occur in between the secondary structure

objects; I do not explicitly model this non-helix, non-strand part of the structure. The secondary structure objects correspond to the core of the protein, the set of residues that tend to be conserved, both structurally and in terms of their amino acid type, within a family of similar structures. The sequence insertions in between the secondary structure objects correspond to the loops which connect the parts of the protein core, and which have been found to be of variable length and composition in a structural family. Finally, I require that every secondary structure object be aligned, in its entirety, to a piece of the sequence. This model of protein structures is used in threading by various researchers [Lathrop *et al.*,].

We can modify the above definitions to incorporate this model of protein structures. Instead of indexing structure elements by j, I will now index secondary structure objects by j. The alignment correspondence function $C(i,j)$ then indicates whether or not secondary structure object j is aligned to the sequence starting at sequence element i. The new match score $M(i,j)$ is the sum of the individual match scores for aligning each element of the secondary structure object j with consecutive sequence elements starting at position i. Because all structure objects must be matched, there is no insertion score I_S.

Computing the optimal alignment

The goal of threading is to find the alignment C that optimizes the score function F. This can be done efficiently, for singleton score functions, using dynamic programming; this approach is described by Needleman and Wunsch [Needleman and Wunsch, 1970]. For a good overview of computational alignment algorithms, see Myers' paper [Myers, 1991]. I will describe the alignment algorithm in this section.

In the alignment algorithm an array, which I will call D, is used to keep track of the scores of optimal partial alignments. An optimal partial alignment C_{mn} is an optimal alignment of the subsequence from positions 1 to m with the secondary structure objects 1 to n, where m and n range from 1 to their maximum values (the sequence length and total number of secondary structure objects, respectively). The nature of the singleton score functions that I use makes it computationally simple to

compute the optimal partial alignment C_{mn} given the optimal partial alignments C_{ij} for $i < m$ and $j < n$.

$D(i,1)$, the score for placing the first structure object at sequence position i, is the sum of the score for the sequence insertion preceding the object, plus the match score for placing the object:

$$D(i,1) = M(i,1) + G(1,i-1),$$

where $M(i,j)$ is the match score for placing secondary structure object j starting at sequence position i, and $G(i_1,i_2)$ is the score for placing an insertion from sequence position i_1 through i_2.

To compute $D(i,j)$ for $2 \leq j \leq N$, where N is the total number of structure objects, we add the match score $M(i,j)$ to the best score over all possible sequence insertion lengths (including 0) for placing the previous element:

$$D(i,j) \;=\; M(i,j) + \max_k \left[D(i - l_{j-1} - k, j-1) + G(i-k, i-1)\right], \qquad \text{for } 0 \leq k \leq \mathcal{L}_{j-1}.$$

Here, l_j is the length of (number of structure positions in) the jth structure object, and $\mathcal{L}_j$ is the sum of the lengths of all structure objects less than or equal to j: $\mathcal{L}_j = \sum_{n=0}^{j} l_n$.

This computation can be sped up by keeping track of the best position of structure object $j-1$ for previous sequence elements, and updating this position with a single call to the gap score function G. This saves computing G for all possible gap lengths at each sequence position.

Once the array D has been computed, the optimal alignment can be found in a trace-back procedure. First we place the last secondary structure object, by finding the sequence position i_N^{opt} which maximizes the expression

$$E(N,M) = D(i,N) + G(i + l_N, M),$$

where M is the length of the sequence and l_N is the length of the Nth structure object. This operation takes into account the gap score for the part of the unmatched part of the sequence on the C-terminal side of the last structure object. The optimal placement of structure object N will be at sequence position i_N^{opt}. The trace-back proceeds by finding the sequence element to optimize $E(j-1, i_j^{\text{opt}} - l_{j-1})$ to place object $j-1$ at an optimal sequence position no higher than $i_j^{\text{opt}} - l_{j-1}$, where i_j^{opt} is the optimal sequence position of object j.

8.1.2 Pseudopotentials for threading

The sequence-structure match scores used in threading are usually based on the ratio of observed to expected frequencies of occurrence, that are an estimate of the likelihood ratio. The counts are of amino acids in given structure environments, or of pairs or triplets of amino acids, again possibly in specified environments. Likelihood ratios, and the ratio of observed to expected counts, have cropped up several times in this thesis. The early work on amino acid preferences for secondary structure types used likelihood ratios, and I used these values as a representation of sequence in the secondary structure prediction chapter. I reported likelihood ratios in the earlier chapters on statistics of amino acid occurrences in protein structures.

The likelihood ratio for two events A and S is the ratio of their joint probability to the product of their individual probabilities (which would be their joint probability were they independent events):

$$L = \frac{P(AS)}{P(A)P(S)}.$$

For example, we can think of $P(AS)$ as the probability of occurrence of an amino acid A in structure S, $P(A)$ as the probability of occurrence of amino acid A, and $P(S)$ as the probability of occurrence of structure category S. When the likelihood ratio differs from 1.0, it indicates that the events are not independent.

These probabilities are estimated by counting the observed frequencies of occurrence in the sample set of known protein structures. For a given amino acid

A and a structure category S, I represent the number of counts as N_{AS}. The number of occurrences of amino acid A is $N_A = \sum_s N_{As}$. The number of occurrences of structure category S is $N_S = \sum_a N_{aS}$. The total number of counts is $N_T = \sum_a \sum_s N_{as} = \sum_a N_a = \sum_s N_s$. Then I estimate the likelihood ratio as the fraction of observed occurrences of amino acid A in structure S, divided by the expected fraction, assuming independence of amino acid type and structure category:

$$\begin{aligned} L &= \frac{(N_{AS}/N_T)}{(N_A/N_T)(N_S/N_T)} \\ &= \frac{N_{AS}N_T}{N_A N_S} \end{aligned}$$

The score for an entire sequence-structure alignment can be computed as the product of the likelihood ratios for each aligned pair of elements. Computationally it is easier to take the logarithm of the likelihood ratios, and sum them. Moreover, the log of likelihood ratios can be related to energies.

8.1.3 Sample size problems

One problem with reporting and using likelihood ratios is that the information about the total number of counts is lost. Low sample size can be a real problem. In earlier chapters, I performed χ^2 tests to determine the significance of a deviation of likelihood ratios from 1.0, the value corresponding to independence of the events measured. I found, for example, that most of the deviations from 1.0 in likelihood ratios measuring pairwise amino acid interactions were not significant.

With an increasing complexity of representation, and a correspondingly smaller number of counts per sequence or structure category, some noise may be introduced that is amplified in the likelihood ratio computation. This noise can be reduced in a number of ways. The basic idea is to keep likelihood ratios closer to 1.0 when there is little data to support more variance. We wouldd like to ensure that the scores that

contribute the most to the overall pseudopotential function have been generated by statistically significant deviations from the expected occurrences of amino acids in given structure environments.

Adding k to numerator and denominator of L.

The simplest way to pad the likelihood ratios is to add a constant to the numerator and denominator, as follows:

$$L_k = \frac{k + N_{AS}N_T}{k + N_A N_S}.$$

The effect of adding a constant offset to the numerator and denominator is to move L toward 1.0. Moreover, larger values of observed counts are less affected than are smaller values of observed counts. Thus, the more data there is, and therefore the more sure we are about our likelihood ratio, the less it is affected by the padding. Figure 8-3 plots the logarithm of the padded likelihood ratio as a function of k, for three different values of $N_{AS}N_T$ and $N_A N_S$. As k increases, $\log(L_k)$ approaches 0.0. For higher numbers of counts, the approach to 0.0 is slower.

Add counts based on independent frequencies.

In this padding paradigm, counts are added to each cell of the table of counts based on the singleton occurrence frequencies of amino acids and structure categories. If the total number of added counts is N_p, then these are distributed according to the singleton frequencies N_A/N_T and N_S/N_T. The number of counts added to each cell is

$$N^p_{AS} = \frac{N_A}{N_T}\frac{N_S}{N_T}N_p.$$

Then the new, padded, cell count is

$$N'_{AS} = N^p_{AS} + N_{AS}.$$

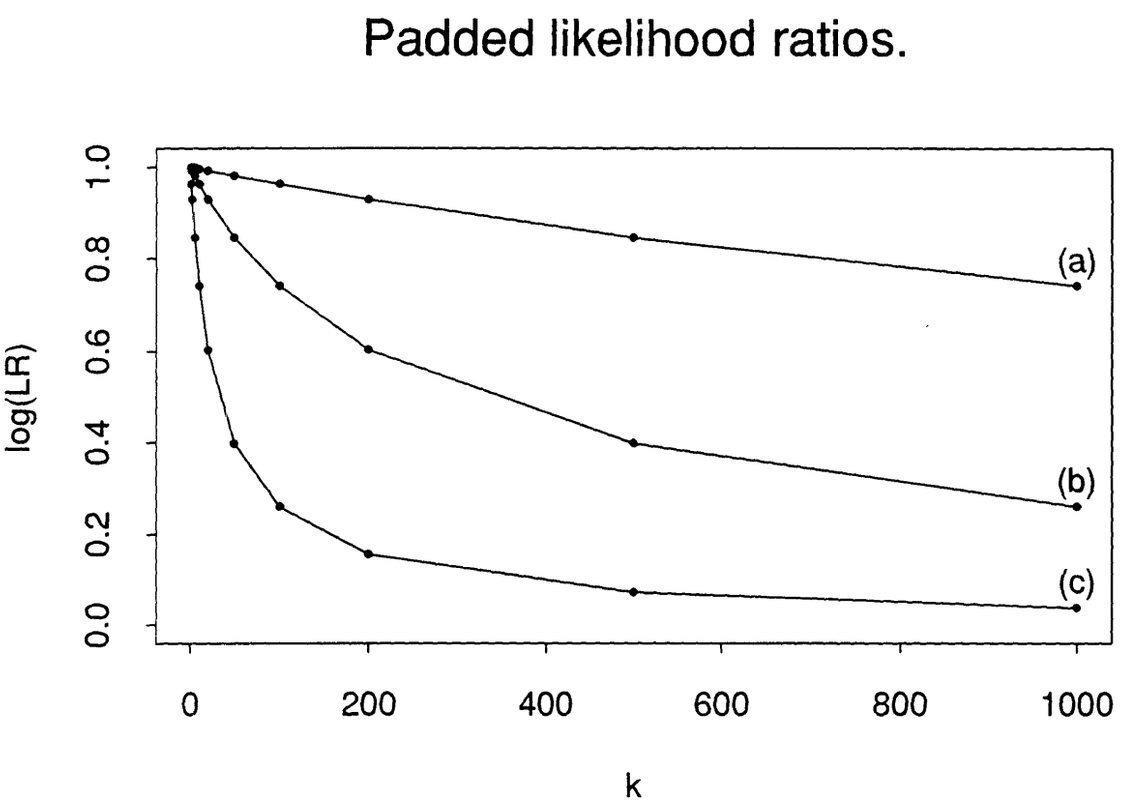


Figure 8-3: Effect of padding the likelihood ratio. $\log(L_k)$ is plotted as a function of k. (a) $N_{AS}N_T = 10,000$ and $N_A N_S = 1000$. (b) $N_{AS}N_T = 1000$ and $N_A N_S = 100$. (c) $N_{AS}N_T = 100$ and $N_A N_S = 10$.

The likelihood ratio is

$$L_{IF} = \frac{N_A N_S N_p + N_{AS} N_T^2}{N_A N_S N_p + N_A N_S N_T.}$$

Comparing L_{IF} to L_k above, we see that where k was added to the numerator and denominator to derive L_k, we are adding $N_A N_S N_p / N_T$ to numerator and denominator. The amount added is dependent on the expected frequency of occurrence of the joint category AS; higher expected frequencies results in greater padding. The philosophy behind this approach, then, is to avoid missing significant interactions that occur between categories with low independent counts.

Add a constant offset to each cell count.

Another possibility is to add a constant offset to each cell in the table of counts; the cell count is now $N_{AS} + k$. Let C_S be the number of different structure (S) categories, and let C_A be the number of different sequence (A) categories. Then the total number of counts is now $N'_T = N_T + kC_A C_S$. The likelihood ratio is

$$L_{CO} = \frac{k^2 C_A C_S + k(C_A C_S N_{AS} + N_T) + N_{AS} N_T}{k^2 C_A C_S + k(N_A C_A + N_S C_S) + N_A N_S}.$$

Thus, to both numerator and denominator is added a constant ($k^2 C_A C_S$) plus a variable term. The numerator's variable term depends on N_{AS}, and the denominator's variable term depends on N_A and N_S.

8.1.4 Structure representations

A structure representation is a set of mutually exclusive categories describing positions or groups of positions in the model structure. For example, the set {alpha, beta, coil} is such a set for singleton positions. For two-element position groups, a complete set would be {alpha/alpha, alpha/beta, alpha/coil, beta/alpha, beta/beta, beta/coil, coil/alpha, coil/beta, coil/coil}, where the first label describes the first residue position, and the second label describes the other residue position.

8.1.5 Incorporating local sequence information

The singleton score functions described above evaluate the preference of one amino acid in the sequence for one structure category. It might be useful to have the match score look at a local piece of sequence around the amino acid residue that is aligned with the structure position. In this case, we are altering the sequence representation. A sequence element, against which a structure element is matched, is now a window of amino acid residues in the sequence instead of a single amino acid residue. There are arguments for and against this approach. On the against side, one might think that the that the complete structure representation we are matching to should provide similar information. Unlike many secondary structure prediction methods, in which the secondary structure of a central residue in a local sequence residue is predicted *in isolation*, threading makes use of local information along the sequence in finding the best alignment of the entire sequence to the entire structure description. On the other hand, there are some reasons that the local sequence information might help. Consider, for example, the fact that the negatively charged residues tend to occur at the N-termini of helices, while positively charged residues tend to occur at the C-termini of helices. The singleton score function described so far, based on the (alpha, beta, coil) structure representation, is not capable of representing this tendency; N- and C-termini of secondary structure objects are not distinguished in the structure models. One solution might be to use local sequence information. In this case, for example, a negatively-charged amino acid residue occuring toward the N-terminus of the central amino acid in the window would support the hypothesis that the central amino acid occurs in a helix object.

This approach is similar to the neural network and GOR (Garnier-Osguthorpe-Robson) methods for predicting secondary structure [Garnier *et al.*, 1978, Gibrat *et al.*, 1987]. Both use a local sequence window. The approach is most similar to the GOR method; this method uses log likelihood ratios to measure the information each amino acid in the sequence window provides toward determining the secondary structure of the central residue. The log likelihood ratios are summed to determine an overall information value for the sequence window. In this chapter, I use sequence

window information in the match function in the same way.

8.2 Method

8.2.1 Data

55 proteins from the Brookhaven database of protein structures were used and are listed in Appendix Section B.2.4, in Tables B.1 and B.2. This list of proteins is a subset of one generated by my colleagues at Boston University [Nambudripad *et al.*,], with 57 proteins. Only non-homologous, monomeric, single-domain proteins solved to high resolution by X-ray crystallography were included. Smaller proteins with very little secondary structure were excluded. The proteins were selected from Release 66 of the Brookhaven protein databank. The list consists of proteins ranging in size from 74 to 405 residues. I eliminate two additional proteins from the list, 1nar and 1ifc. 1nar does not have an HSSP file in the Sander and Schneider database, while the secondary structure for 1ifc is mislabeled in the Kabsch and Sander DSSP file.

Homology derived secondary structures files were obtained for each of the 55 proteins from the Schneider and Sander HSSP database [Sander and Schneider, 1991], by ftp from the European Molecular Biology Laboratory. These files contain lists of similar sequences aligned to the sequence for which there is a known structure.

8.2.2 Threading code

I wrote code in Lisp and C to implement the dynamic programming threading algorithm for singleton pseudopotentials, to compute the score functions, and to display the results of threading.

8.2.3 Structure representations

I tried various structure representations. Figure 8-4 shows the structure representations that I used. For example, the representation Exp-SS-coil contains five categories: alpha buried, alpha exposed, beta buried, beta exposed, and coil. When "coil" is part

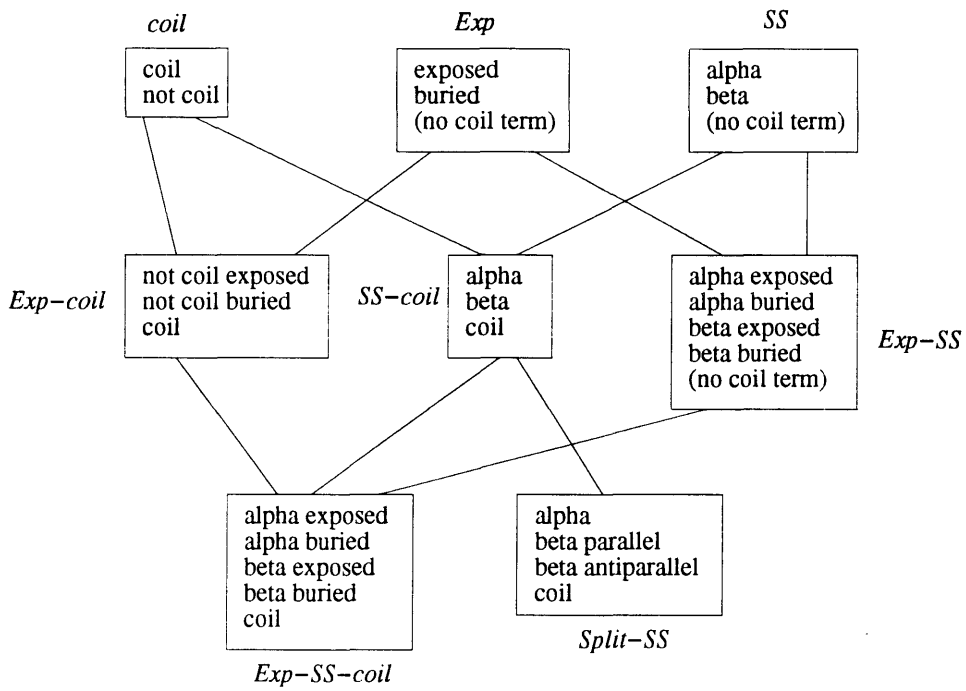


Figure 8-4: Structure representations. Each box shows one representation, and includes a list of the structure categories used in that representation. The lines indicate the hierarchy of representations; upper boxes are connected by a line to boxes containing representations that are a refinement obtained by splitting categories.

of a structure representation, that indicates that an insertion score is assigned to each sequence element placed in the gaps between secondary structure objects. When coil is not part of a structure representation, there is no score assigned to parts of the sequence that fall between secondary structure objects. The remaining structure categories are assigned to residue positions within the structure objects.

I used the Kabsch and Sander definitions of secondary structure to determine the secondary structure objects. Any helix ("H" label in the Kabsch and Sander system) or strand ("E" label) consisting of two or more residues became a secondary structure object. The structure category "alpha" refers to residues in a helix object and the structure category "beta" is given to residues in a strand object. The structure category "exposed" applies to residues with relative solvent exposure (as measured by Kabsch and Sander) of greater than 0.29; "buried" applies to all other residues. "Beta parallel" residues are any beta strand residues involved in a parallel strand-strand

ladder. "Beta antiparallel" residues are all other beta strand residues. Note that if a residue is involved in both a parallel and antiparallel ladder, it is classified as parallel. This is based on the assumption that the more restrictive sequence requirements for parallel sheets will take priority.

8.2.4 Amino acid counts

The score functions were created from counts of the number of occurrences of amino acid residues in the different structure categories. For the purpose of creating the score functions, the amino acids were classified as follows:

- Any residue labeled "H", "G", or "I" in the Kabsch and Sander scheme was considered helix.
- Any residue labeled "E" or "B" was considered beta.
- All other residues were considered coil.

The exposure definitions were the same: relative exposures above .29 indicated an exposed residue; exposures below were buried.

The aligned sequence data compiled by Sander and Schneider was used to augment the data set. For any residue position, there is a set of aligned residues, one from each of the aligned sequences. At every position, each different amino acid type was considered as a separate count. For example, if a position contained the residues `IIILIHLVVVYI`, and if it were in the "beta buried" structure category, then the counts in the following cells of the table of counts would each be incremented by 1: (I, beta buried), (L, beta buried), (H, beta buried), (V, beta-buried), (Y, beta buried).

8.2.5 Scores for threading; padding

Scores were computed as described in Section 8.1.2. In some experiments, padding by adding a constant offset to both numerator and denominator of the likelihood ratio was used.

8.2.6 Incorporating local sequence information

Log likelihood ratios L^n_{AS} were determined for the joint occurrence in the set of known-structure proteins of an amino acid A in residue $i+n$ and secondary structure category S at residue i, where n is 0 or a small positive or negative integer between $-K$ and $+K$. Various values of K were tried. The log likelihood ratios were added to obtain the match score. The score of the central residue was emphasized by weighting it by an amount W. The overall match score is therefore

$$M(i,j,K,W) = (W-1)\log(L^0_{AS}) + \sum_{n=-K}^{+K} \log(L^n_{AS}).$$

Various values of local window size parameter K and central residue weight W were tried. Both SS-coil and Exp-SS-coil structure representations were used.

8.3 Results and Discussion

8.3.1 Counts

Table 8.1 shows the counts for the Exp-SS-coil structure representation. There are 39,683 total counts. Trp (W) is the amino acid with the fewest occurrences (472); Ala (A) is the amino acid with the most occurrences (3075). Strand exposed structure is the most rare, with 3200 counts. The other three helix and strand all have counts of about 6300. There are 17,432 coil counts.

8.3.2 Likelihood ratios

Table 8.2 gives the likelihood ratios for the Exp-SS-coil structure representation, computed from the count data in table 8.1. For example, Phe (F) shows a preference for buried structure, a preference against exposed, and avoids coil. Ala (A) shows a preference for alpha structure, and a preference to avoid beta and coil structures.

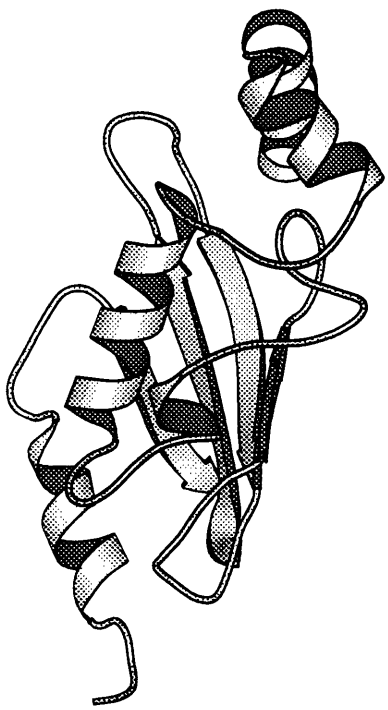
8.3.3 Comparison of singleton pseudopotential components

AA	Abur	AExp	BBur	BExp	Coil	Total
A	642	557	458	198	1220	3075
C	174	107	182	78	840	1381
D	218	495	187	175	1190	2265
E	312	631	213	236	1002	2394
F	368	116	433	82	497	1496
G	268	283	268	122	1326	2267
H	151	176	161	99	484	1071
I	524	161	624	140	638	2087
K	267	609	218	283	1175	2552
L	696	274	630	156	888	2644
M	296	115	248	63	337	1059
N	223	418	197	187	1198	2223
P	139	225	133	113	851	1461
Q	270	470	198	201	803	1942
R	210	410	213	201	823	1857
S	345	506	405	259	1436	2951
T	352	395	405	277	1163	2592
V	562	225	739	194	841	2561
W	107	52	108	34	171	472
Y	256	115	311	102	549	1333
Tot.	6380	6340	6331	3200	17,432	39,683

Table 8.1: Counts for Exp-SS-coil structure representation. AA: amino acid. ABur: alpha buried. AExp: alpha exposed. BBur: beta buried. BExp: beta exposed. Coil: not alpha or beta.

AA	ABur	AExp	BBur	BExp	Coil
A	1.30	1.13	0.93	0.80	0.90
C	0.78	0.48	0.83	0.70	1.38
D	0.60	1.37	0.52	0.96	1.20
E	0.81	1.65	0.56	1.22	0.95
F	1.53	0.49	1.81	0.68	0.76
G	0.74	0.78	0.74	0.67	1.33
H	0.88	1.03	0.94	1.15	1.03
I	1.56	0.48	1.87	0.83	0.70
K	0.65	1.49	0.54	1.38	1.05
L	1.64	0.65	1.49	0.73	0.76
M	1.74	0.68	1.47	0.74	0.72
N	0.62	1.18	0.56	1.04	1.23
P	0.59	0.96	0.57	0.96	1.33
Q	0.86	1.51	0.64	1.28	0.94
R	0.70	1.38	0.72	1.34	1.01
S	0.73	1.07	0.86	1.09	1.11
T	0.84	0.95	0.98	1.33	1.02
V	1.36	0.55	1.81	0.94	0.75
W	1.41	0.69	1.43	0.89	0.82
Y	1.19	0.54	1.46	0.95	0.94
Ave.	1.03	0.95	1.04	0.98	0.98
S.d.	1.71	1.67	2.06	1.03	1.17

Table 8.2: Likelihood ratios for Exp-SS-coil structure representation. AA: amino acid. ABur: alpha buried. AExp: alpha exposed. BBur: beta buried. BExp: beta exposed. Coil: not alpha or beta. Ave.: column averages. S.d.: column standard deviations.



1aak

150 residues

ubiquitin conjugating enzyme

Figure 8-5: Diagram of protein 1AAK, drawn by the program Molscript.

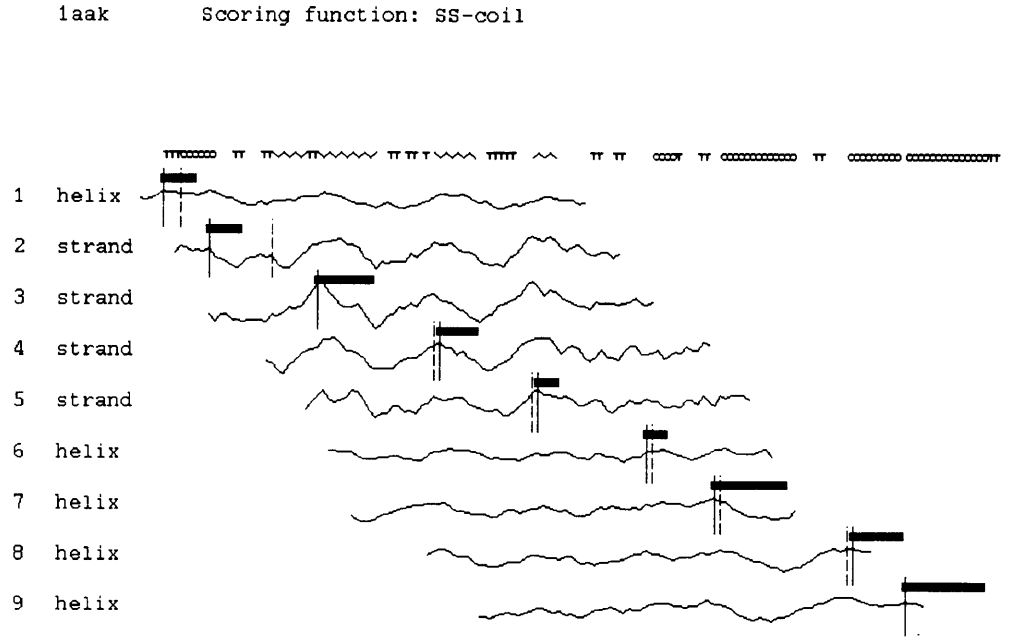


Figure 8-6: Threading of 1aak with SS-coil pseudopotential. The top line of symbols shows the Kabsch and Sander secondary structure: a circle is helix, zig zag lines are strands, Ts represent turns. Each secondary structure object in the structure is shown on a separate line. For each, a trace of the match score of the element to the sequence starting at position i is shown, as a function of i, for all legal values of i for that secondary structure object. A solid vertical bar shows the optimal threading; a dashed vertical bar shows the correct threading.

I illustrate the results of the threading program on the protein 1AAK, which is pictured in Figure 8-5; this sketch was drawn by the program Molscript [Kraulis, 1991]. Figures 8-6, 8-7, and 8-8 show the results of running the threading code on protein 1AAK for three different pseudopotentials: SS-coil, Exp-coil, and Exp-SS-coil.

The top line of symbols in each figure shows the secondary structure of the protein as assigned by Kabsch and Sander using the DSSP program. A circle represents a helical residue, a diagonal line represents a strand, and a T represents a turn.

There is one plot for each secondary structure object below the secondary structure labeling line. Each plot, or trace, shows the match score $M(i, j)$ for object j along the sequence i (i increases from left to right in the diagram). The plots are scaled to fit the display area, but the relative magnitudes are preserved between structure objects.

Because all the structure objects must be placed, each score trace starts no further to the left than that object can be placed and still fit the preceding objects. Similarly, the score trace ends on the right to leave enough room to fit all the remaining structure objects.

A solid vertical line marks the optimal position of the object, as computed by the threading algorithm. This usually occurs at a local minimum of the score function, as we'd expect. A dashed vertical line marks the correct position of the object. The dashed line does not appear in the diagram when the two are superimposed. At the top of each optimal vertical line there is a vertical box which shows the length of the object. For structure representations which include the exposed/buried distinction, this box is coded so that white represents exposed residues, and black represents buried residues.

Figure 8-6 shows the 1AAK threading for the SS-coil structure representation, which is composed of the three categories alpha, beta, and coil. The first structure object in 1AAK is a helix, and therefore the first plot shows $M(i,0) = \sum_{k=i}^{i+L(0)-1} F(A_k, \alpha)$, where $F(A,S)$ is the (log-likelihood-ratio) score for matching amino acid A to structure category S, $L(0)$ is the length of the first object (6), and A_k is the kth amino acid in the sequence. Note the broad peaks and valleys in the score as the sequence objects scan the sequence. This is not surprising because the structure category along the entire structure object is the same and we are in effect taking a running average of the α or β score in a subwindow along the sequence. If two of the secondary structure objects were of the same type (alpha or beta) and length, their traces would be exactly the same, though starting and ending at different sequence positions.

For this protein, the SS-coil score manages to place most of the objects in approximately the right positions.

The score traces in Figure 8-7 (the Exp-coil structure representation) stand in sharp contrast to those of Figure 8-6: they have a much higher frequency; this reflects the changing pattern of structure categories along each structure object. The beta strand objects have patterns of alternating exposure which are reflected in the score trace. The slower period of the characteristic amphipathic frequency of the helices is

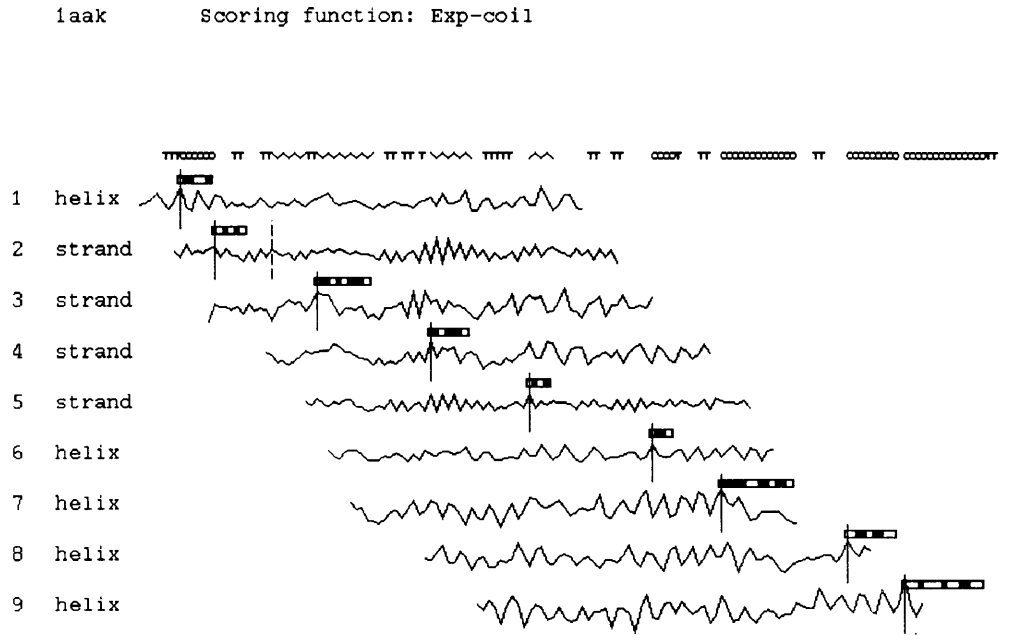


Figure 8-7: Threading of 1aak with Exp-coil pseudopotential.

also apparent in the helix score traces.

The Exp-SS-coil score function is qualitatively very similar to that of Exp-coil. Both Exp-coil and Exp-SS-coil place eight of the nine objects correctly.

Table 8.3 shows the results for threading on the set of 55 proteins. There are 680 secondary structure objects. The table shows the percentage of objects correctly placed, either exactly, or within two or four sequence elements. Distinguishing between alpha and beta (SS-coil, 20.3%) does not improve the results much over the simpler coil/not-coil representation (Coil, 17.6%). Distinguishing between buried and exposed (Exp-coil, 43.5%) is a much better structure representation than one based on secondary structure type. This is a striking confirmation of the predictions of the statistical analysis described in previous chapters. A structure representation that incorporates both solvent exposure and secondary structure (Exp-SS-coil, 49.4%) performs somewhat better than using solvent exposure alone. Again, this confirms the statistical analysis results.

In Table 8.4, I show results for other structure representations; in particular,

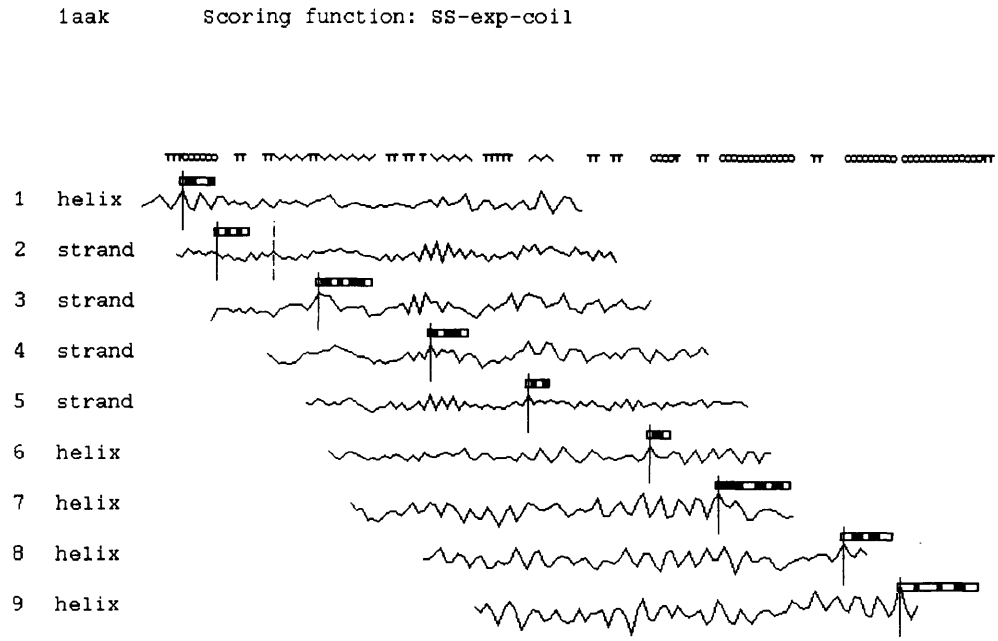


Figure 8-8: Threading of 1aak with Exp-SS-coil pseudopotential.

Expmt.	NC	All			Alpha			Beta		
		0	±2	±4	0	±2	±4	0	±2	±4
Coil	2	17.6	51.2	60.3	15.1	51.6	64.5	19.5	50.9	57.4
SS-coil	3	20.3	52.4	63.2	17.6	52.3	65.2	22.2	52.4	61.8
Exp-coil	3	43.5	59.9	72.6	54.1	64.2	81.0	36.2	56.9	66.8
Exp-SS-coil	5	50.1	68.7	79.7	59.9	70.6	84.9	43.4	67.3	76.1

Table 8.3: Results for singleton experiments. Numbers in table are percentages of secondary structure objects exactly placed (0), placed within two sequence elements of the correct placement (±2), and placed within four sequence elements of the correct placement (±4). Results are shown for all secondary structure objects, and for alpha and beta objects separately. NC is the number of categories in the structure representation.

Expmt.	NC	All			Alpha			Beta		
		0	±2	±4	0	±2	±4	0	±2	±4
Coil	2	17.6	51.2	60.3	15.1	51.6	64.5	19.5	50.9	57.4
SS	2	9.0	25.0	35.4	4.7	18.6	32.6	12.0	29.4	37.4
SS-coil	3	20.3	52.4	63.2	17.6	52.3	65.2	22.2	52.4	61.8
Exp	2	37.1	52.5	64.6	47.3	55.9	60.3	29.9	50.1	60.6
Exp-coil	3	43.5	59.9	72.6	54.1	64.2	81.0	36.2	56.9	66.8
Exp-SS	4	48.2	64.9	75.1	58.1	67.4	81.0	41.4	63.1	71.1
Exp-SS-coil	5	50.1	68.7	79.7	59.9	70.6	84.9	43.4	67.3	76.1

Table 8.4: Further results for singleton experiments.

I compare the results with and without the use of insertion scores for the regions of sequence that are not matched to secondary structure objects. Using a score function in these loop regions gives moderate improvement in performance, although this improvement is not great for the most successful Exp-SS structure representation.

8.3.4 Incorporating sequence window information

Table 8.5 shows the percent of correctly placed secondary structure objects for various combinations of the window length parameter K and the central residue weight W. A sequence window size of 3 residues ($K = 1$), with a center multiplier W of 5, produced the best results, two percentage points higher than the base case with no local sequence information. The full results for this best case are shown in Table 8.7, in comparison to the results without local sequence information.

When I use the structure representation Exp-SS-coil, I do not find improvement in the threading results in the combinations of K and W that I tried (Table 8.6). The alpha and beta results are shown for the best K, W combination in Table 8.7. I have not found a way to exploit any additional information the local sequence might provide over and above that available in the Exp-SS-coil structure representation and single-residue likelihood ratios.

	K				
W	0	1	2	3	6
1	20.3	17.5			13.8
2					17.1
5		22.4	21.5	21.4	20.0
10		21.5	21.3	20.9	21.9
15					21.0
50					20.7
100					20.6
1000					20.4

Table 8.5: Results for incorporating local sequence information in SS-coil representation. Numbers in table are percent correctly placed secondary structure objects, out of a total of 680. K is the window length parameter and W is the weight on the central residue's score.

	K				
W	0	1	2	3	6
1	50.1	43.8	41.9	37.8	
5		49.4	49.0		
10		49.4	48.5	48.5	
20		50.7	49.3	50.1	
50		50.4	50.0	50.0	49.7

Table 8.6: Results for incorporating local sequence information in Exp-SS-coil representation. Numbers in table are percent correctly placed secondary structure objects, out of a total of 680. K is the window length parameter and W is the weight on the central residue's score.

SR	NC	K	W	All			Alpha			Beta		
				0	±2	±4	0	±2	±4	0	±2	±4
SS-coil	3	0	1	20.3	52.4	63.2	17.6	52.3	65.2	22.2	52.4	61.8
SS-coil	3	1	5	22.4	52.6	63.2	19.7	52.7	66.3	24.2	52.6	61.1
Exp-SS-coil	5	0	1	50.1	68.7	79.7	59.9	70.6	84.9	43.4	67.3	76.1
Exp-SS-coil	5	1	20	50.7	69.7	80.4	59.9	71.7	85.3	44.4	68.3	77.1

Table 8.7: Results for local sequence window experiments. Best result and results without local sequence information are shown for both SS-coil and Exp-SS-coil structure representations. Numbers in table are percentages. "SR": structure representation. "NC": number of count categories. "K": sequence window parameter. "W": multiplier on central residue.

8.3.5 Splitting the beta structure representation into parallel and antiparallel

I expected that breaking up the beta structure categories into parallel and antiparallel would improve the threading results because the likelihood ratios for the two types are quite different for some residues (Table 8.9). The amino acid preferences for and against parallel structure appear to be stronger than those of antiparallel, alpha, or coil structure. His and Thr even change preference, favoring antiparallel and disfavoring parallel structure.

However, I did not find consistent prediction improvement; in fact, the split structure representation, Split-SS-coil, performed slightly worse than the two-category Coil representation! When compared to SS, the structure representation with no coil, the Split-SS representation did slightly better (9.7% as opposed to 9.0%), particularly on placing beta strands.

It seemed likely that the problem was the relatively low number of samples, particularly in the case of the parallel structure, and the therefore possibly misleadingly high magnitudes of the corresponding scores. The standard deviation for the parallel strand likelihood ratios is much higher than that of the other structure types (Table 8.9). A high standard deviation within a structure category could be an indication either that the structure representation is good at discriminating between amino acids, or that there is a low sample size. Therefore, this seemed like an ideal case in which to apply the low-sample-size solutions of padding. The results for various values of the padding constant k are shown in Table 8.10. There is an improvement as k increases, up to an optimal value for k at around 10^8, which is about 10 times the average value of the denominator of the likelihood ratio (1.6×10^7 for $N_T = 39,683$). The results for the Coil and SS-coil representations are shown for comparison. With padding, I am able to obtain, with the split beta representation, threading results slightly better than those for the unpadded SS-coil representation.

This result suggests that low sample size is a relevant problem, and that padding may help to alleviate it. So far I have been dealing only with singleton structure rep-

AA	Alpha	Par	Ant	Coil	Total
A	1199	120	513	1220	3052
C	281	42	203	840	1366
D	713	53	289	1190	2245
E	943	60	368	1002	2373
F	484	107	391	497	1479
G	551	53	318	1326	2248
H	327	36	215	484	1062
I	685	191	538	638	2052
K	876	69	405	1175	2525
L	970	183	569	888	2610
M	411	62	238	337	1048
N	641	53	313	1198	2205
P	364	33	203	851	1451
Q	740	34	348	803	1925
R	620	56	338	823	1837
S	851	102	537	1436	2926
T	747	101	552	1163	2563
V	787	208	680	841	2516
W	159	32	105	171	467
Y	371	60	338	549	1318
Tot.	12,720	1655	7461	17,432	39,268

Table 8.8: Counts for Split-SS.

AA	Alpha	Par	Ant	Coil
A	1.21	0.93	0.88	0.90
C	0.64	0.73	0.78	1.39
D	0.98	0.56	0.68	1.19
E	1.23	0.60	0.82	0.95
F	1.01	1.72	1.39	0.76
G	0.76	0.56	0.74	1.33
H	0.95	0.80	1.07	1.03
I	1.03	2.21	1.38	0.70
K	1.07	0.65	0.84	1.05
L	1.15	1.66	1.15	0.77
M	1.21	1.40	1.20	0.72
N	0.90	0.57	0.75	1.22
P	0.77	0.54	0.74	1.32
Q	1.19	0.42	0.95	0.94
R	1.04	0.72	0.97	1.01
S	0.90	0.83	0.97	1.11
T	0.90	0.94	1.13	1.02
V	0.97	1.96	1.42	0.75
W	1.05	1.63	1.18	0.82
Y	0.87	1.08	1.35	0.94
Ave.	0.99	1.03	1.02	1.00
s.d.	0.71	2.35	1.06	0.93

Table 8.9: Likelihood ratios for Split-SS, along with averages and standard deviations within each structure category.

k	All			Alpha			Beta		
	0	±2	±4	0	±2	±4	0	±2	±4
0	17.1	46.9	57.8	14.7	48.4	62.4	18.7	45.9	54.6
10^6	18.1	49.4	60.3	15.1	49.8	64.2	20.2	49.1	57.6
10^7	20.7	54.6	65.0	17.9	55.9	69.2	22.7	53.6	62.1
10^8	21.9	53.5	63.3	20.1	53.4	65.5	23.2	53.6	61.7
10^9	21.0	52.2	62.2	20.4	51.6	63.1	21.4	52.6	61.6
(Coil)	17.6	51.2	60.3	15.1	51.6	64.5	19.5	50.9	57.4
(SS-coil)	20.3	52.4	63.2	17.6	52.3	65.2	22.2	52.4	61.8
(SS)	9.0	25.0	35.4	4.7	18.6	32.6	12.0	29.4	37.4
(Split-SS)	9.7	27.2	37.6	4.3	18.3	32.3	13.5	33.4	41.4

Table 8.10: Improvement of threading performance by padding the singleton scores for the Split-SS representation. Numbers in table are percentages. The results for the Coil, SS-coil, SS, and Split-SS representations are shown for comparison.

resentations; structure representations with more categories, such as those modeling pairwise interactions between residues, are likely to suffer much more from the low sample size problem.

8.4 Conclusions

I draw the following conclusions from the self-threading experiments:

- Solvent exposure preference is more useful than secondary structure preference for structure prediction.
- Secondary structure may add information above and beyond the solvent exposure preference, at least for a crude exposed/buried solvent exposure model.
- The coil / not-coil distinction is more useful than the alpha / beta distinction for structure representation.
- Modeling loop regions in the structure prediction is helpful.
- Incorporating sequence information improves prediction when the secondary structure representation is used; however, I have not found it to be helpful in improving prediction beyond that obtainable by using solvent exposure.

- Unless problems of low sample size are resolved, splitting the beta strand category into parallel and antiparallel does not improve results.
- Low sample size can be ameliorated by padding the likelihood ratios used as matching scores.

Chapter 9

Work in Progress

9.1 Computer representations of proteins

There are many further areas for exploration.

9.1.1 Automation

It might be possible to automate the statistical analyses, so that model exploration relevant to the choice of protein representation features can be done on the computer. There exist heuristics for choosing appropriate model hierarchies that could provide a starting point [Goodman, 1971].

I have chaperoned the analysis of a given protein representation through the statistics and test application programs. It would be neat to automate this process. A protein representation evaluator would be given the protein representation (or a set of protein representations to be compared) as input, and would have access to the database of known structures, and possibly other information about proteins (Figure 9-1). Statistical and application tests would be run, and a report would be generated. The report might include recommendations on which elements of the representation were most important, and avenues to try next in improving the representation.

At the next level of automation, I imagine a protein representation generator (Fig-

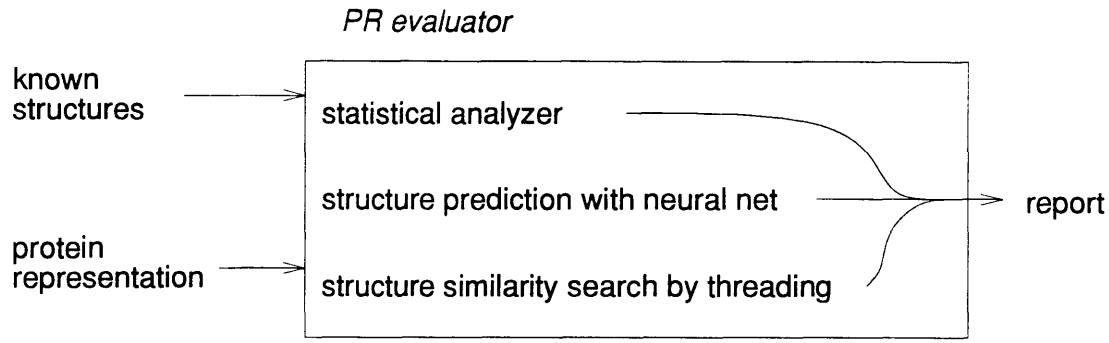


Figure 9-1: Automatic evaluation of protein representations (PRs).

ure 9-2). This program would incorporate the protein representation evaluator as a component. There would also be a component that would modify a given representation in ways suggested by the evaluator's report. These modifications would include the operations I tried by hand in the threading chapter: generalizing, specializing, and adding or dropping new attributes. In addition, there would be one component of the system that would intelligently explore the database of known structures with a goal of creating entirely new, and potentially useful, attributes. This component would also have access to other relevant information about proteins.

9.1.2 Statistical analysis

I want to extend the statistical analysis to other types of topological pairs, and explore ways of reducing the complexity of the protein representation while maintaining the important features.

The results of the statistical analysis I have done in this thesis should be fed back into further analyses of protein structures. Interesting questions have been raised which deserve being pursued by looking, for example, at the geometry of pairs of residues.

It would be useful to mathematically extend the contingency table theory to allow table cells to be joined if they behave similarly, thus producing more statistical significance (and therefore generalization power) while allowing important single cells

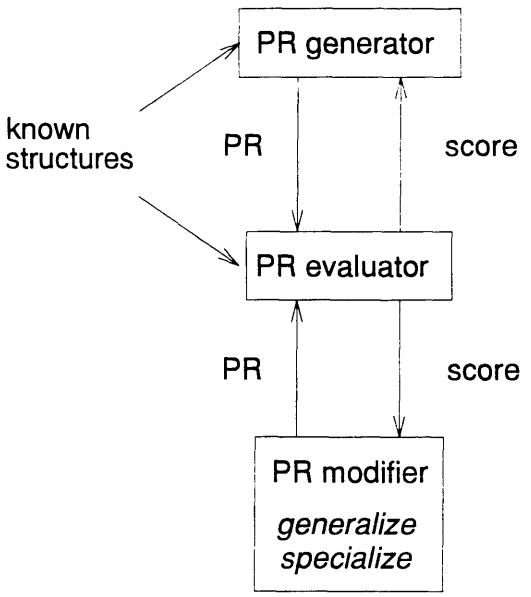


Figure 9-2: Automatic generation of protein representations (PRs).

to remain as they are.

9.1.3 Threading

In the threading work, I have looked only at single-residue properties. Threading should be done on residue-pair properties as well as on the properties particular topological relationships in beta sheets and other structures.

9.1.4 Other representations

There are many representations, both entirely new approaches and variations on the ones that I have tried, that could be explored.

For example, other definitions and representations of local structure exist. One such representation is a local window along the amino acid chain, with atomic positions stated for all the atoms in the backbone of the protein. This window can be seen as a building block for the protein.

Unger *et al.* (1989) clustered segments of length six from a database of proteins, and used representative segments from each cluster as their representation of sec-

ondary structure. They found that they could build proteins using these blocks with some degree of accuracy. They have not yet tried predicting the secondary structure from primary sequence, but the technique looks promising. Some of the representative segments corresponded to traditional secondary structure elements such as helices, but others were new, and potentially useful.

Ideas for sequence and structure representations should come from analyses of similar structure, such as that by Flores and colleagues [Flores *et al.*, 1993].

9.2 Solvent exposure

A recurring theme in this thesis is that solvent exposure is important. How can we use that fact? I find that there are some buried beta sheets that are amphipathic. Could this be a requirement for folding? Can we use this information in predicting protein structure?

In this section, I give several examples of buried amphipathic proteins, then talk about how amphipathicity might be incorporated into the threading algorithm.

9.2.1 Buried polar beta sheet faces

I show here several buried amphipathic sheets.

Rhodanase (1rhd)

Rhodanase (1rhd) is a transferase enzyme 293 residues long. It contains a buried parallel beta sheet with one face that is clearly polar; in particular, one buried string of beta pair partners is NDTYK.

Figure 9-3 shows the structure of rhodanase. The figure was drawn by the program Molscript [Kraulis, 1991]. There are two subdomains. I will focus on the sheet in the top subdomain, of which three strands are shown in this picture (the edge strands, having only two residues, were not displayed using the arrow notation). The sheet is completely surrounded by alpha helices.

Figure 9-4 is a diagram of the layout of the sheet. There are five strands, each of

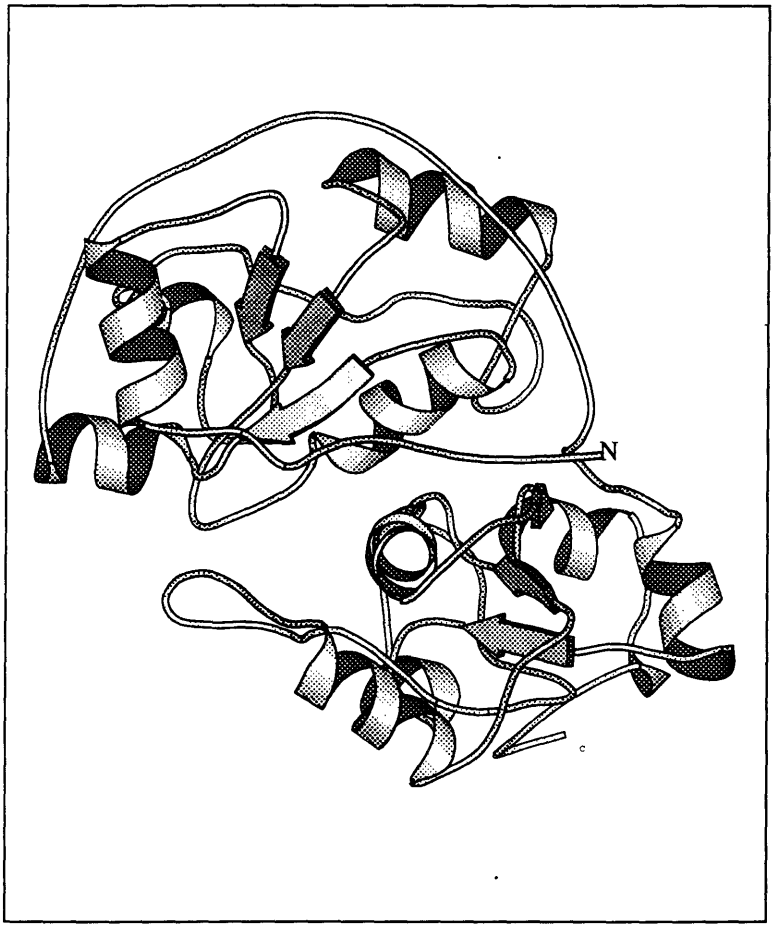


Figure 9-3: Molscript drawing of rhodanase, 1rhd.

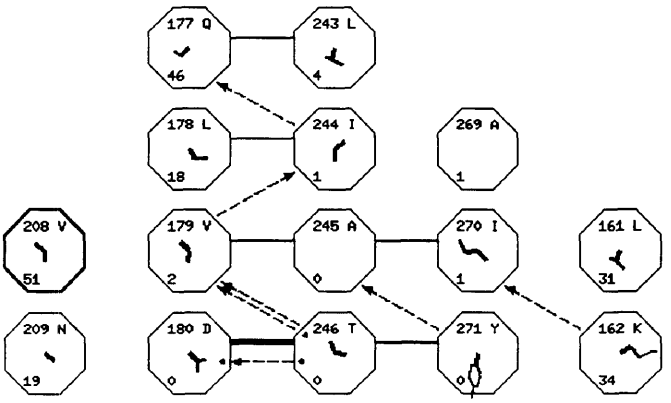


Figure 9-4: One sheet of rhodanase, 1rhd. See text for description of symbols

which runs vertically in the diagram. Each octagon corresponds to a residue, and is labeled with the residue number (top left), the amino acid type (top right), and the solvent exposure (bottom left). In the middle of the octagon is a two-dimensional projection, down the C_α–C_β vector, of the bonds connecting the non-hydrogen atoms in the residue's side-chain. The thickness of the octagon perimeter is monotonically related to the residue's relative exposure (fraction of maximum exposure for that amino acid type). Solid lines show beta-pair sidechain contact; thicker lines indicate more contact area. Dashed lines represent hydrogen bonds; the arrow points from the donor toward the acceptor. A dot at the tail or head of the hydrogen-bond arrow indicates that the hydrogen bond originates or terminates at a side-chain atom; if there is no dot, then the hydrogen bond is at the backbone. The important thing to note is that the bottom row of residues (NDTYK) is polar and buried.

Figure 9-5 is a stereogram of one face of the sheet, showing the line of buried hydrophilic residues.

Elastase (3est)

Elastase (structure 3est) is a digestive enzyme with sequence similarity to trypsin and chymotrypsin. The sequence identity is higher for residues in the interior of the enzyme. All three proteins have similar three-dimensional structures, and a serine-histidine-aspartate catalytic triad (residues 102, 57 and 195 in elastase, which occur in loops). The catalytic mechanism involves enabling a tetrahedral transition state and the formation of a covalent acyl-enzyme intermediate.

The three-dimensional structure of elastase includes two antiparallel beta barrels, or orthogonally packed beta sheet pairs, packed against each other (Figure 9-6). The barrels define two subdomains of the protein; the interface between these subdomains is crossed three times by the backbone, although two of these crossings are near the two ends of the protein.

Figure 9-7 diagrams the outer face of one of the barrels. This face contains many polar residues. Some of the polar residues are exposed to solvent, as indicated by the DSSP-computed accessibility numbers in the figure. However, some of the residues

Figure 9-5: Stereogram of one face of 1rhd sheet. The line of buried hydrophilic residues is at the bottom.

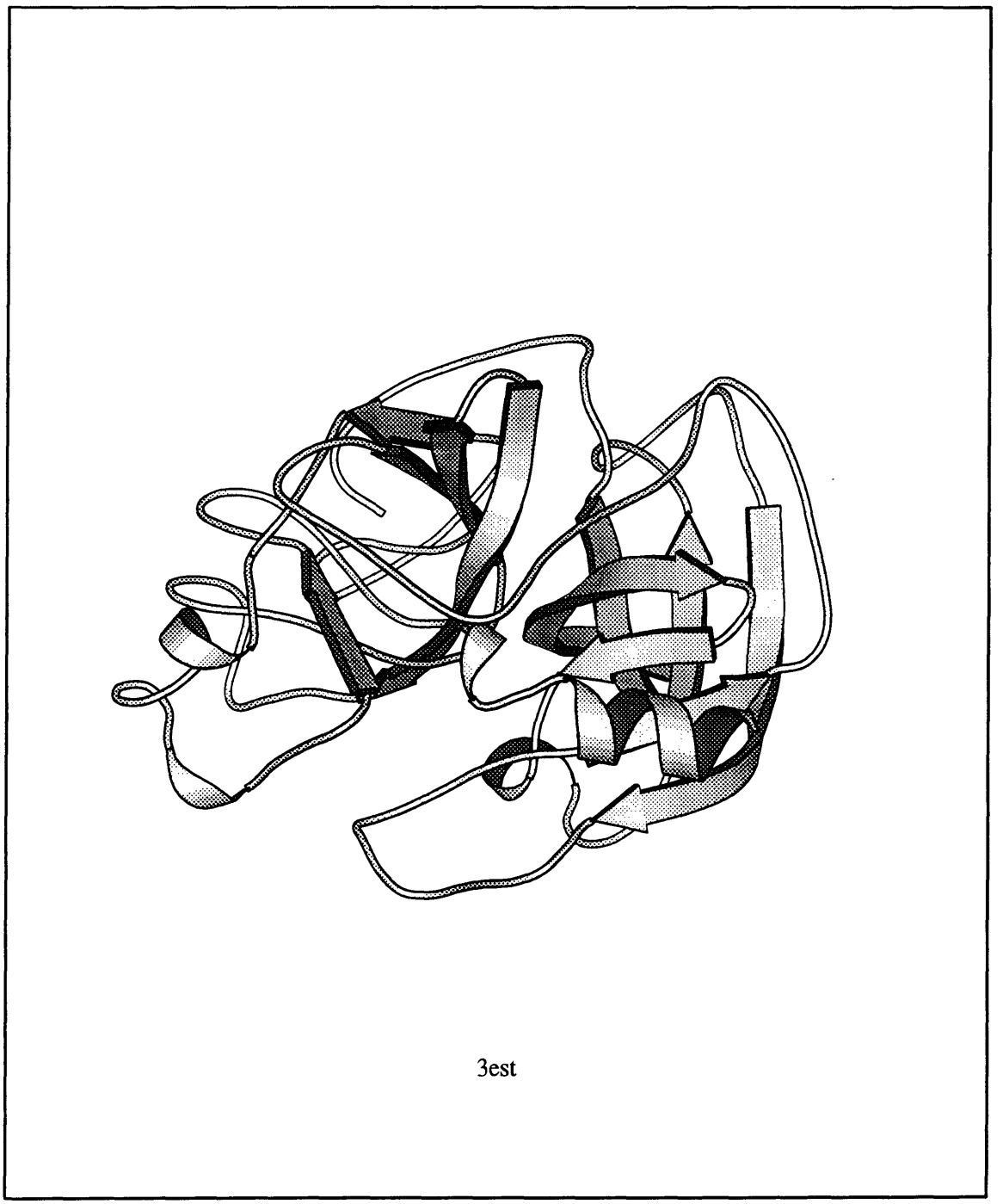


Figure 9-6: Structure of elastase.

are quite buried, notably 30Q (accessibility 6), 45T (accessibility 0), 32S (accessibility 0), 34Q (accessibility 11), 40H (accessibility 17), 53H (accessibility 0), 51H (accessibility 4). These residues are situated at the interface between the two structural subdomains.

Porin

Porin is a 16-stranded antiparallel beta barrel membrane channel protein [Kreusch *et al.*, 1994]. Porin is inside out with respect to most beta barrels. Its outer surface sees the hydrophobic environment of the lipid bilayer. Through the center of the molecule runs an aqueous pore. The alternating hydrophobic character of the strands is very clear, as shown for a couple of strands in the excerpt from an HSSP file in figure 9-8. Note the smattering of glycines which help reduce the strand twist for the long strands.

9.3 Amphipathic Models for Threading

How can we include the amphipathicity constraint in the core domain threading method? In this section I describe some preliminary work I did on incorporating amphipathicity in the definition of the structure model for threading. The results on a few proteins seem promising, but more work is needed to determine whether this approach might bear fruit. I describe here the preliminary results I obtained.

Each residue in the model structure is labeled buried or exposed. The singleton term in the score function is computed for each residue by looking up the score for that amino acid in the environment defined by the exposure (buried or exposed) and secondary structure (alpha or beta). There are various ways to determine the solvent exposure of a model position. A threshold is applied to this exposure number to determine whether the site is buried or exposed.

To encourage amphipathicity in the secondary structure elements, I tried two approaches. The first was to manually search in exposure label space to find labels which gave good threadings. In fact, I quickly found optimal threadings for the two

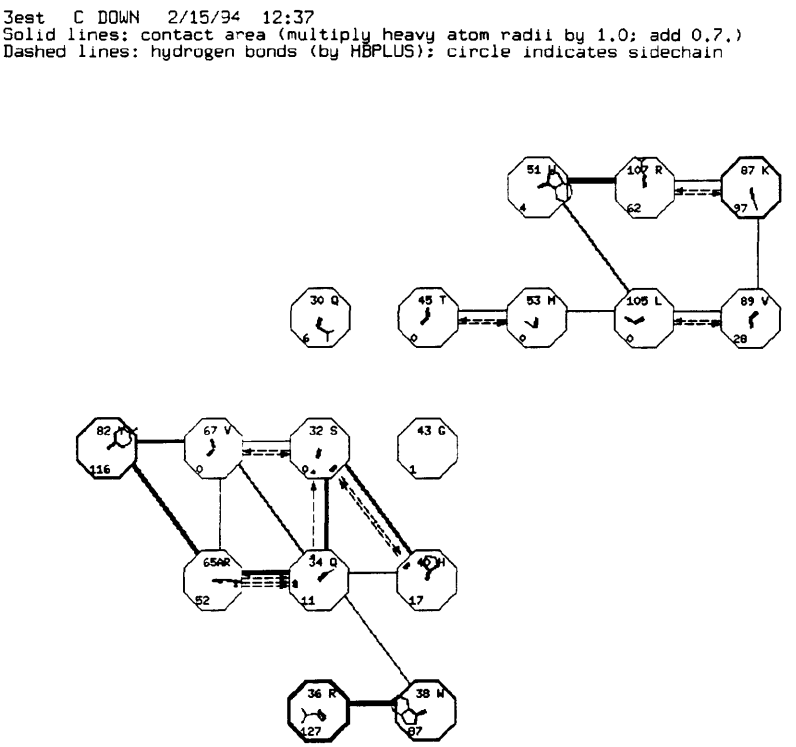


Figure 9-7: 3est, one side of sheet.

```
pdbno   AA   exposure aligned seqs.
 40     Y      33       YYYYYYYYYY *
 41     I      60       IIVVVAAAMM
 42     R      35       RRRRRRRRRR *
 43     F      87       FLFFFLLLLI
 44     G       0       GGGGGGGGGG *
 45     F      88       FFFIIFFFFF
 46     K      86       KKKKKKKKKK *
 47     G      20       GGGGGGGGGG
 48     E      85       EEEEEEEEEE *
 49     T      57       TTTTTTTTTT

 55     L      40       LLLLLLLLLL
 56     T      13       TTTTTTTTTT *
 57     G       7       GGGGGGGGGG
 58     Y      46       YYYYYFFYYY *
 59     G      18       GGGGGGGGGG
 60     R      58       RRRRRQQQQQ *
 61     W      67       WWWWWWWWWW
 62     E      16       EEEEEEEEEE *
 63     A       0       AAASSYYYYY
 64     E      23       EEEEEEENQQ *
 65     F      75       FFFFFFFFII
 66     A      30       AAASSKKQQQ *
```

Figure 9-8: Excerpt from the Porin hssp file. Those positions marked with an asterisk correspond to residues whose side chains point inward toward the pore.

Figure 9-9: Amphipathic exposure patterns for strands and helices. Black squares represent buried residues, and white squares represent exposed residues.

proteins I worked with.

The second approach was less biased: I used separate thresholds for buried and exposed residues, and allow an intermediate range in which the amphipathicity constraint takes hold. Any positions with exposure less than threshold θ_b are labeled buried. Any positions with exposure greater than threshold θ_e are labeled exposed. Any positions with intermediate values of exposure are labeled with an amphipathic pattern. The pattern is constrained by the labeled residues (if any) on either end. If there is still ambiguity in how to place the amphipathic pattern over residues, then I choose the labeling that maximizes a function that evaluates how well the pattern fits the exposure values in the unlabeled window. This compatibility function has the form

$$F_n = \sum_{i=l}^{r} A(i) L_n(i),$$

where l is the leftmost residue index in the unlabelled window; r is the rightmost residue index in the unlabelled window; $A(i)$ is the solvent accessibility of the i'th residue; n indexes the offsets of the amphipathic pattern, and $L_n(i)$ is a function indicating for offset n whether the i'th residue is buried or exposed. For an $A(i)$ in which higher values represent greater exposure, L takes the value 1 for exposed and -1 for buried residues in the pattern.

The amphipathic pattern for beta strands is strictly alternating buried and exposed residues. The amphipathic pattern for alpha helices corresponds to a helix of period 3.6 residues.

9.3.1 Perfect Self-Threading on trypsin inhibitor (1tie) and pseudoazurin (2paz)

As a first test of the potential for incorporation of an amphipathic constraint in structure models for threading, I selected two proteins that contain beta structure, pseudo-azurin (2paz) and trypsin inhibitor (1tie). The proteins are illustrated in figures 9-10 and 9-11. For each protein, I manually changed the exposure labels and with very little effort, was able to achieve perfect self-threading. In order to do this, I looked at scans, across the whole sequence, of the threading score for each secondary structure element. I also examined the DSSP solvent accessibility numbers. The resulting patterns were more amphipathic.

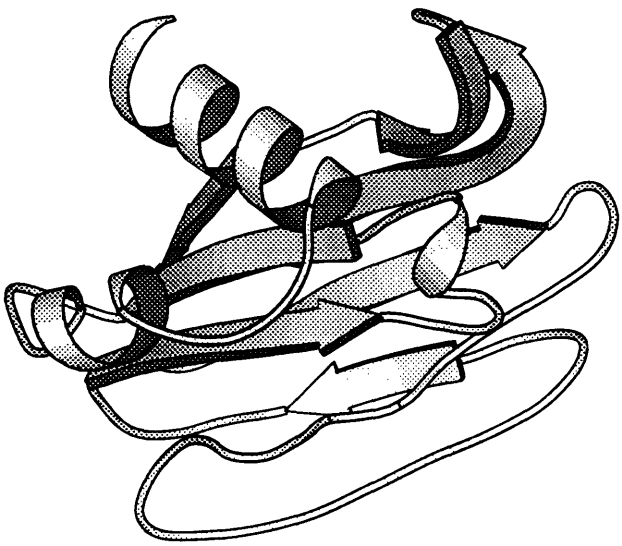
The optimized labels for 2paz also did better on the homologous sequences than did the original labels.

9.3.2 Threading Sequence-Homologous Proteins

The fact that I was able to get much-improved exact self-threading by only changing the exposure values was an encouraging sign. I next proceeded to see whether similar sequences would show improved threading. In fact, I find that a model incorporating the amphipathic constraint performs better on a set of sequences similar to pseudoazurin (structure 2paz) than a model whose exposure labels were determined based on a single threshold. The number of correctly placed segments went from 71/181 (39%) to 106/181 (59%).

The full model for 2paz contains nine strands and two helices. However, inspection of the aligned helices in the HSSP file shows that most aligned sequences are missing those two helices. Therefore, I created a reduced model containing only the nine beta strands. Some of the aligned sequences do not align to all nine strands.

The *original* exposure labelings in the model were obtained by thresholding the relative accessibility (obtained by dividing the DSSP accessibility by the maximum accessibility for that amino acid type) at 0.29. The *amphipathic* exposure labels were obtained by requiring that all of each secondary structure be strictly amphipathic.



2paz

Figure 9-10: Structure of pseudoazurin.

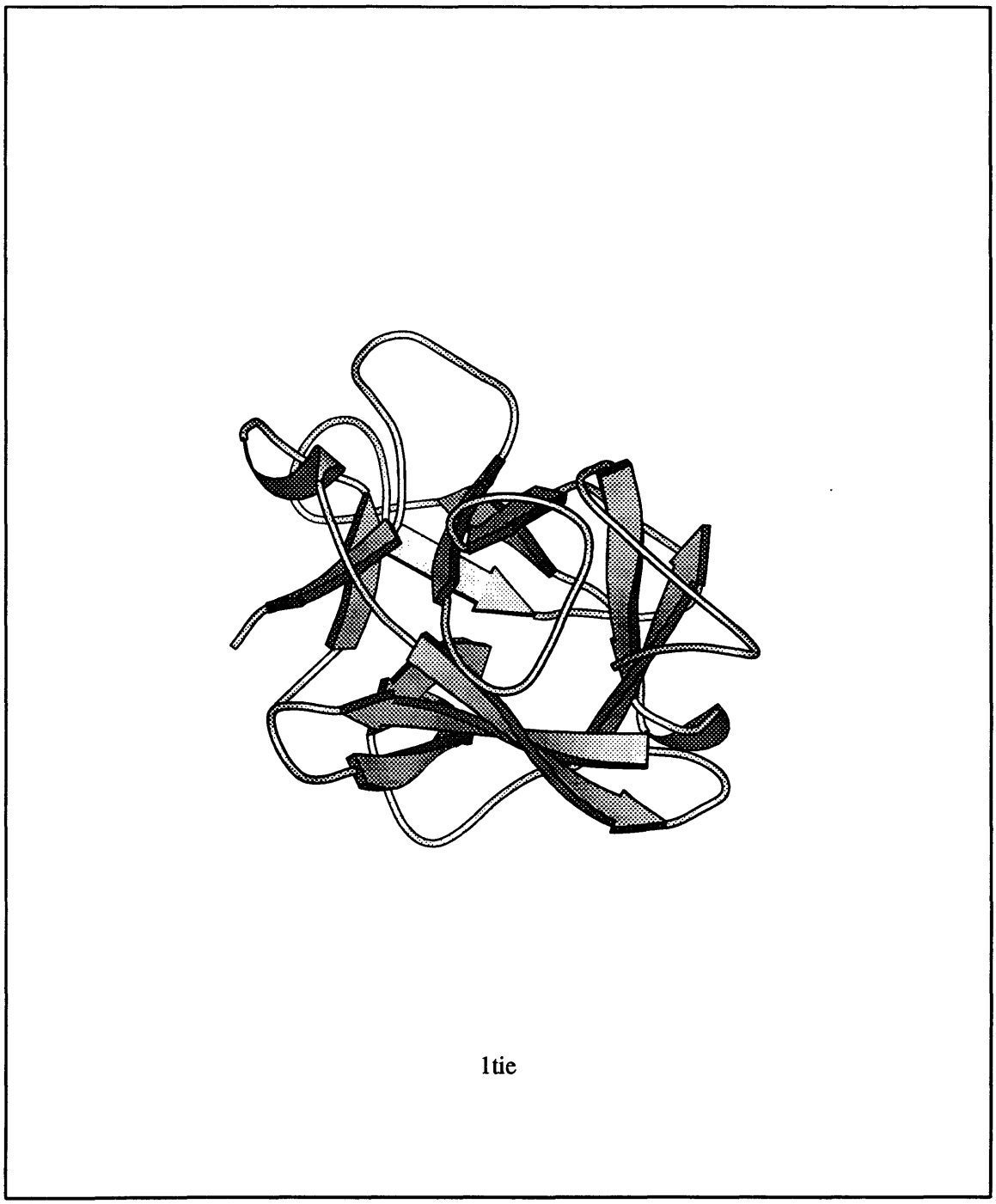


Figure 9-11: Structure of pseudoazurin.

Exposure	Orig.	Amphi.	Optimal
2paz 9	4 6 6	1 1 1	6 7 9
azup_alcfa 9	1 1 1	1 1 1	1 1 2
azup_vibal 9	5 8 9	7 8 9	7 8 9
azup_metex 9	4 6 6	0 1 1	4 6 6
amcy_parde 7	0 0 1	0 0 0	0 0 0
plas_anasq 8	1 1 1	1 1 1	1 1 1
plas_anava 8	4 6 6	3 3 3	5 5 5
plas_orysa 8	2 3 3	2 2 2	4 5 6
plas_sceob 8	6 7 7	3 5 6	4 4 4
plas_lacsa 8	6 8 8	5 8 8	7 8 8
plas_entpr 8	3 6 6	3 3 3	5 8 8
plas_ulvar 8	4 7 7	2 2 3	5 7 8
plas_samni 8	2 5 5	5 8 8	6 8 8
plas_chlfu 8	5 7 7	3 7 7	5 7 7
plas_horvu 9	2 2 2	0 0 0	3 4 5
plas_petcr 8	7 8 8	5 7 8	7 7 8
plas_merpe 8	2 5 5	3 3 3	6 8 8
plas_phavu 8	2 5 5	3 3 3	8 8 8
plas_soltu 8	2 3 3	3 3 3	5 8 8
plas_vicfa 8	4 7 7	3 4 4	5 8 8
plas_spiol 9	1 1 1	1 1 1	5 6 6
plas_dauca 8	4 5 5	5 7 8	7 7 8
total exactly right	71	59	106

Figure 9-12: In the results shown in the figure, three numbers are reported for each threading of a sequence against the nine-stranded structure model. The first number represents the number of secondary structure segments placed exactly correctly; the second is the number of segments placed within $\pm$ 2 of the correct placing, and the third is the number of segments within $\pm$ 4 of the correct placing.

The *optimal* exposure labels were obtained by manually changing exposure labels on some positions to enhance the amphipathicity of the model elements. The optimal model places more segments exactly correct than does the original model, and more often places all segments approximatly correctly (within 4 residues). The strict amphipathic constraint fares the worst.

How would an unbiased assignment rule do? A two-threshold rule (0, 90) performs worse than the others (64), but with a score function incorporating information from homologous sequences, it does better (83) and gets all of the segments correctly placed within 4 residues for most (11) of the sequences. This is compared to the model run with the same score function, but based on a sequence-independent ("geometric") accessibility, which gets 71 segments correctly placed, and places segments within 4 residues on only 5 of the sequences.

9.3.3 Two-Threshold Rules for Labeling Model Residue Exposures

We compared the single-threshold rule to a two-threshold rule on a set of 46 proteins. To compute accessibilities, all sidechains were replaced by Alanine residues (with CB radius 2.1), and the DSSP program was run to compute accessibility with a water radius of 2.4 Angstroms. The default exposure labeling was obtained by using a threshold of .29 on relative accessibility. Scores were computed using a set of aligned sequences, leaving out the protein of interest. Dynamic programming was performed to align the model with the sequence using singleton scores only, including gap scores. The default model placed 48% of the segments correctly, 72% within 2, and 81% within 4.

A two-threshold model with thresholds of .2 and .4 on the relative exposure performed slightly better, with 49% of the segments placed exactly correctly, 73% within 2, and 82% within 4.

Pascarella and Argos [Pascarella and Argos, 1991] report that only 38% of beta strands show a hydrophobic moment peak in the beta frequency range, and only 9%

are conserved across aligned sequences.

I use aligned sequences to enhance the amphipathicity signal in proteins.

The hydrophobic moment of a window of residues at a given frequency ω is computed as in Cornette et al. [Cornette *et al.*, 1987]:

$$\mu(\omega) = \frac{2}{L}\left|\sum_{k=0}^{L-1}(h_k - h_a)\mathrm{e}^{ik\omega}\right|^2$$

L is the length of the window; h_k is the hydrophobicity of the k'th residue in the window; h_a is the average hydrophobicity in the window. The amphipathic index for a given frequency range is computed (following Cornette) as

$$\mathrm{AI}(\omega_1, \omega_2) = \frac{\frac{1}{x}\sum_{\omega=\omega_1}^{\omega_2-1}\mu(\omega)}{\frac{1}{180}\sum_{\omega=0}^{\omega=179}\mu(\omega)}$$

I used the PRIFT hydrophobicity scale computed by Cornette et al.

The amphipathicity index, AI, was computed for 10° intervals of the 180° range, for window sizes corresponding to beta (5, 7) and alpha (11, 13, 15) conformation. For each window size, the location of the peak AI was recorded. Following [Pascarella and Argos, 1991], I expect beta conformation to have a peak between 150° and 180°, and alpha conformation to have a peak betweeen 80° and 110°.

For each protein, the AI peaks were computed in two ways. First, a single sequence was used to find the peaks. Then, the peaks were computed using aligned sequences as follows: in each aligned position, the most hydrophilic amino acid was chosen. This is similar to the approach used by Thornton et al., who take the most hydrophilic amino acid after discarding the most hydrophilic amino acid (to make room for error in the sequence alignment). The justification for this is based on the fact that interior residues tend to be conserved and hydrophobic, while exterior sites generally show a lot of variation and can accommodate both hydrophilic and hydrophobic amino acids. Moreover, given that a protein is only marginally stable, it is likely that while the hydrophobicity patterns may not be apparent from looking at a single sequence, the pattern of substitutions across an aligned set may show the hydrophobicity pattern.

For proteins containing exposed beta sheet, using aligned sequences gives a clear

improvement in finding amphipathic beta strands. Figures 9-13 through 9-15 shows this improvement for pseudoazurin sequences. Each row in the figure represents one position in the protein. The dssp sequence number, amino acid, and secondary structure (using the Kabsch and Sander DSSP notation) are listed first. The next 5 columns show the results for a hydrophobicity analysis for the single protein sequence of 2paz. The first two of these columns correspond to the expected beta lengths, and the last three are the expected alpha lengths. In a cell, a + is marked if the hydrophobic moment peak for the window centered on that residue occurs in a region corresponding to that column's secondary structure, and the residue does indeed have that secondary structure. A − is marked if the residue does not have that secondary structure, and the hydrophobic moment peak indicates that there is secondary structure. Thus, a + or − in a column labeled 5 or 7 indicates a hydrophobic peak in the range $150° - 180°$; + indicates that the residue is in beta conformation and − indicates that the residue is not in beta conformation. Many of the − labels occur at residues flanking beta structure, which indicates that the hydrophobic pattern extends beyond the region labeled beta by the DSSP program.

There are no windows of length 7 that contain all beta residues. There are 8 windows of length 5 that contain all beta residues, centered at residues 16, 17, 32, 33, 74, 75, 89, and 90. All 8 windows have a hydrophobicity moment in the $170° - 180°$ range, using the aligned sequence data. Using only the 2paz sequence, only one of the 8 windows has a hydrophobicity moment in the beta range, at $150° - 160°$.

There are several windows of length 11 and greater at the C-terminal end of 2paz which contain all alpha residues, and the amphipathicity of these is found both by the single-sequence and the multiple-sequence methods.

Residue			2paz sequence					Aligned sequences					
			Beta		Alpha			Beta		Alpha			
No.	AA	SS	5	7	11	13	15	5	7	11	13	15	aligned sequences
8	L	E						+	+				LLL L L LKL LLLLL
9	N	E						+	+				NNN L L LLL LLLLL
10	K	E						+	+				KKS G G GGG GGGGG
11	G	E						+					GGG A SGGGAA SSGAG
12	A	T											AKP N SDDEDN DGDSG
13	E	T											EDG KKGSDDDDSGDDDDDDD
14	G	E							+				GGG GGGGGGGGGGGGGGGG
15	A	E						+	+				AAM LLVAGSASAVGESSGSA
16	M	E						+	+				MMMMLLLLLLLLLLLLLLL
17	V	E						+	+				VVVKVVVVVAAAVVVAVAAAV
18	F	E						+	+				FFFYFFFFFFFFFFFFFFF
19	E	E						+	+				EEDEEEEEEVVIEESVVIVLS
20	P	S											PPPTPPPPPPPPPPPPPPP
21	A	S											AAAPAANASNSSSNSNSGNGS
22	Y	E						+	+				YSLEKKDTTNKNSDSNENSDS
23	I	E						+	+				ILVLLLFVFIIFVFFFFFFF
24	K	E						+	+				KKRHTTTTSTSSTSTSSSESS
25	A	E					−	+	+				AVLVIIVIVVVVIVVVVVVV
26	N												NAKKKKKKAGAPKKAPPSSAA
27	P	T											PPPVPPSASAASAAASSAASK
28	G	T	−										GGGGGGGGGGGGGGGGGGG
29	D												DDDDDDEDEEEEEEEEEDEE
30	T	E											TTSTTTTSKSAKTTKKKKTEG
31	V	E							+				VVIVVVIVIIIIVIIIIIII
32	T	E						+	+				TTKTEETTVEETTTTTVTVVS
33	F	E						+	+				FFFWFFFWFFFFWFFFFFFF
34	I	E						+	+				IILILLKTKIVKVKKKKKKKK
35	P	E	+										PPPNNNNNNNNNNNNNNNN
36	V	S											VTTRNNNNNNNNNNNNNNN
37	D	S		−									DDDEKKAAAAAAAAAAAAAA
38	K	T											KKKavvgggggggggggggg
39	G	T											GGGpppppppppppppppp
40	H												HHHHHHHHHHHHHHHHHHH
41	N		−					−					NNNNNNNNNNNNNNNNNNN
42	V								−				VVVVVVVIVIIVIVIVVVVI
43	E	E											EEEHVVVVVVVVVVVVVVV

Figure 9-13: 2paz amphipathicity, part 1 of 3. + indicates correct label; − indicates false positive. "No.": dssp number. "AA": amino acid. "SS": secondary structure.

Residue			2paz sequence					Aligned sequences					
			Beta		Alpha			Beta		Alpha			
No.	AA	SS	5	7	11	13	15	5	7	11	13	15	aligned sequences
44	S	E			−								STTFFFFFFFFFFFFFFFFF
45	I				−	−							IIIVDDDDDDDDDDDDDDDD
46	A	T			−	−	−						KKKAAAEEEEEEEEEEEEEE
47	D	T			−	−	−						DGGGTADDDDDDDDDDDDDD
48	M	S			−	−	−						MMMVLLAAEAAEEAEEEEEE
49	I	S			−	−	−						IIALNNVVIVVVVVVIIIIIV
50	P				−	−	−			−			PPPGPPPPPPPPPPPPPPPP
51	E	T			−	−	−			−			EDDEAASAAAASSSASAASSA
52	G	T			−	−	−			−	−	−	GGGAKKGGGGGGGGGGGGGG
53	A				−	−	−			−	−	−	AAAASSVVVVVVAVVVVVVVV
54	E					−	−		−		−	−	EEDLAADNDDDDNDNDDDDDD
55	K						−		−		−	−	KAYKDDVAAAASAVAAAAAAV
56	F		−	−		−	−	−		−		−	FFVGllSDSDDAESESVSAAS
57	K	B	+	+			−	+			−		KKKpkkkakaakakkkkkkk
58	S		−	−			−	−					SSTmSSnnnnnnntnnnnnnd
59	K		−	−		−			−				KKTKLLAAASSAAAGAAAAAG
60	I	T		−				−					IIVKSSPPPKKPPPAPPAPPA
61	N	T											NNGEHHGGGGGGGGGGGGGG
62	E												EEQQKKEEEQEEEEEEEEEE
63	N				−								NNEAQQTSTTTTSTTTTTTS
64	Y	E							+				YYAYLLFYYVVYYFYYYYYYF
65	V	E						+	+				VKVSLLSTAVVSSSEAVSSKT
66	L	E		+				+	+				LVVLMMVAVRRVAVVVVVVVV
67	T	E	+					+	+				TTKTSSTKTKKTKTTTTTKTT
68	V								−				VFFFPPLFLLLLFLLLLLLL
69	T			−				−	−				TTDTGGTDTTSTDTTTDSDTT
70	Q	S							−				QAKEqqVTETTETVEETEAEE
71	P							−	−				PPEAaaPAKPPSAPKKKKKK
72	G	E						+	+				GGGGGGGGGGGGGGGGGGGG
73	A	E						+	+				AVVTDETETTVTTTTSTTTTT
74	Y	E						+	+				YYYYYYYYYYYYYYYYYYYY
75	L	E						+	+				LGGDSTGGSGGKGGKSSTKKK
76	V	E		+				+	+				VVFYFFFYFVVFYFFFFFFFF
77	K	E						+	+				KKKHYYYFYYYYFYYYYYYYY
78	C		−		−			−	−				CCCCCCCCCCCCCCCCCCCC
79	T	T		−									TTATEEEEADESEEESSASSE

Figure 9-14: 2paz amphipathicity, part 2 of 3. + indicates correct label; − indicates false positive. "No.": dssp number. "AA": amino acid. "SS": secondary structure.

Residue			2paz sequence					Aligned sequences					
			Beta		Alpha			Beta		Alpha			
No.	AA	SS	5	7	11	13	15	5	7	11	13	15	aligned sequences
80	P	T											PPPPPPPPPPPPPPPPPPPP
81	H	T		−									HHHHHHHHHHHHHHHHHHHH
82	Y	G											YYYPRRAQQSAQQAAQQQQQA
83	A	G	−	−				−					AGMFGGGGGGGGGGGGGGGG
84	M	G	−	−									MMMMAAAAAAAAAAAAAAAA
85	G	T						−					GGGRGGGGGGGGGGGGGGGG
86	M							−	−				MMM MMMMMMMMMMMMMMM
87	I	E						+	+				IVV VVVVVKKVKVKVVVVVK
88	A	E						+	+				AGA GGGGGMMGGGGGGGGG
89	L	E						+	+				LVL KKKKKTTKTKEKKKQKE
90	I	E						+	+				IVV IIVVVIIVIVVVVVVVV
91	A	E						+	+				AEV TTTITTTTTTTTTTTTT
92	V	E							+				VVV VVVVVVVVVVVVVVVV
93	G	S											GGG AAN N N NNNNNNN
94	D	S											DDD SG
95	S	S	−										SAK
96	P		−	−			−			−			PPR
97	A	T		−	−	−	−			−			AAD
98	N	T	−		−	−	−						NNN
99	L	H			+	+							LLL
100	D	H			+								DEE
101	Q	H					+						QAA
102	I	H											IVA
103	V	H			+		+						VKK
104	S	H				+	+						SGS
105	A	S					−						AAV
106	K						−						KKQ
107	K												KNH
108	P									−			PPN
109	K	H					+					+	KKK
110	I	H					+					+	IKL
111	V	H					+					+	VAT
112	Q	H			+	+	+				+		QQQ
113	E	H			+	+	+			+	+		EEK
114	R	H			+	+	+			+	+	+	RRR
115	L	H			+	+	+				+	+	LLL

Figure 9-15: 2paz amphipathicity, part 3 of 3. + indicates correct label; − indicates false positive. "No.": dssp number. "AA": amino acid. "SS": secondary structure.

Chapter 10

Conclusions

Knowledge representation for protein structures

There are many ways to represent protein structure. I've looked at a number of these representations in the context of developing methods for predicting protein structure. I've considered secondary structure representations (alpha, beta, coil), representing pairwise information, local sequence information surrounding a particular residue, amino acid hydrophobicity and patterns of hydrophobicity along a sequence. These representations have various levels of complexity. For example, the category of beta strand secondary structure could be split into parallel and antiparallel strand. I've considered residue pairs be defined in a number of ways, based on side-chain contact or topological relationships within secondary structure pieces. There have been many representations of structure and sequence proposed that I have not touched on.

The knowledge representation that we choose shapes the information that we gather and use in modeling and predicting. For the various kinds of information that we might want to represent about a protein, there are a number of questions we might ask. How redundant are the various types of information? How do we use the representations? How do we combine different types of information? How do we choose which representation we want?

I want to use the knowledge representations that I've been studying in protein structure prediction. In particular, I'd like to make use of the knowledge representa-

tions in looking at the set of known protein structures, and through the lens of these representations gather information about the properties of proteins that could then be used in predicting structure of other proteins.

The problem of insufficient sample size

One problem that I run into is that using a complex, fine-grained knowledge representation inhibits my ability to make conclusions about protein properties. This happens because there isn't enough data for me to be able to make conclusive statements about every category in my representation. These complex protein representations can result in inferior protein structure prediction methods.

So what is there to do about the low sample size problem?

One approach is to use all the data that we have, and to use it in ways that are as clever as possible. In addition to using as many known-structure proteins as we can, for example, we might also try to use information from aligned sequences. This is an approach that a number of people have taken. I use it in this thesis in determining scores for threading functions.

Another approach is to incorporate a notion of uncertainty into the use of empirical data. The data about which we are less certain should have less influence on the prediction. In the thesis I have discussed ways of padding the data to improve the prediction results.

Even using all the data that we have, and accounting for the certainty with which we know it, we are still likely to run up against problems with sample size for complex knowledge representations. At this point, we can try to reduce the complexity of the knowledge representation. And this is where the work I've done in the thesis becomes interesting.

Making hard choices in knowledge representation

So the problem is to explore the space of possible representations of protein structure and sequence in order to find one which is not too complex but which still suits our purposes in enabling good structure prediction. If we are going to choose a simple,

compact representation in which we keep only elements which are the most important to the problem at hand, we need to have ways of deciding which elements to keep and which to throw out. It might not be enough to find all the important factors, because some of these might be redundant with each other. It is also important to keep in mind the way we use the knowledge representation. A representation might be very useful with one prediction algorithm but perform poorly with another.

In this thesis, I present methodologies for evaluating structure and sequence representations. First, I use contingency table analysis to determine what information is important in representing structures, and how it is related. In addition, this statistical analysis gives me some idea of the relative importance of different components of the protein representation. However, this statistical analysis is performed without regard to the particular algorithms in which the representations are to be used. Therefore, I also use another approach, which is to incorporate the knowledge representations in structure prediction algorithms, and to compare them by seeing how well they perform in the predictions.

The methodologies that I use are not limited to the particular knowledge representations or structure prediction methods that I've used. There are many ways to represent local secondary structure, for instance, other than by the (alpha, beta, coil) distinction.

Solvent exposure is important

In my experiments I discovered that the most important factor (of the ones I examined) in structure prediction is the exposure to solvent of a residue in a protein. The types of amino acids that occur on the outside of proteins are very different than the ones that occur most often on the inside of the proteins. While this is a generally known fact, I showed that this effect is much more powerful than, say, the preference of amino acids for a particular type of local secondary structure. This suggests that in the course of developing a knowledge representation for protein structure, it's likely to be a good idea to skimp on the secondary structure in favor of a good solvent exposure representation, given the choice. In the thesis, I show that effects related to

solvent exposure can even be useful in predicting secondary structure.

The importance of amino acid hydrophobicity and solvent exposure in determining the structure of proteins is not all that surprising in light of current models of how a protein folds. The hydrophobic collapse model of folding holds that the driving force in protein folding is the segregation of hydrophobic residues to the inside of the protein, and hydrophilic residues to the outside. This fact is used by many researchers in the field of protein structure prediction. For example, many pseudopotentials for threading contain terms related to the solvent exposure of the residues.

On the other hand, there are many approaches which do not consider solvent exposure. It might be worth looking at these and asking whether it is possible to use the same algorithms on a representation that includes solvent exposure.

Appendix A

Related Work

A.1 Counting atom and residue occurrences in protein structures

A.1.1 Single-residue statistics

There are many studies of single-residue statistics in the literature; I include only a few relevant examples here.

There exist several studies of the amino acid composition of secondary structure.

Amino acid composition of secondary structure: Chou and Fasman looked at 4,741 residues in 29 proteins [Chou and Fasman, 1978], and used the observed secondary structure preferences to classify the residues for their rule-based secondary structure predictor.

Levitt tabulated the frequencies of amino acids in alpha helix, beta strand, and reverse turn [Levitt, 1978].

Lifson and Sander counted the occurrences of each type of amino acid in parallel and antiparallel beta sheet. They found that the compositions of parallel and antiparallel sheets were different from each other.

The Garnier Osguthorpe Robson method involves using an information theoretic approach to extract information about the amino acids which occur in a window around a given type of secondary structure [Garnier *et al.*, 1978, Gibrat *et al.*, 1987]. The frequencies of occurrence have also been tabulated by amino acid property [Kelley and Holladay, 1987].

Wertz and Scheraga looked at the preferences of amino acid residues for the inside or outside of the protein [Wertz and Scheraga, 1978].

Patrick Argos and Jaume Palau [Argos and Palau, 1982] divided beta structure into categories according to the position of the residue in the strand relative to the N- and C-termini. For each position, they examined the amino acid composition. They found considerable asymmetry in the properties of residues at different positions along the strand. Ala and Gly prefer to occur in the middle of long strands; Cys is preferred in the middle of (any length) strands.

Others have found that dividing into finer classifications was useful. Wilmot and

Thornton [Wilmot and Thornton, 1988] analyzed beta turns by classifying them into 3 categories (I, II, and nonspecific) and deriving separate Chou-Fasman-like preferences for each class. They found that this improved secondary structure prediction (using a Chou-Fasman-like algorithm). McGregor, Flores and Sternberg [McGregor *et al.*, 1989] used neural networks to predict these same turn classes, and found better turn prediction but worse assignment of predicted turns to their correct class.

Cid and colleagues found that different amino acids have different apparent hydrophobicity as they occur in proteins of different structural classes [Cid *et al.*, 1992].

A.1.2 Pair interactions

Von Heijne and Blomberg [von Heijne and Blomberg, 1978] counted pairs of residues in specific topological relationships in beta sheets. They grouped residues into three classes: hydrophobic, neutral, and polar. They considered three kinds of pairs: interstrand neighbors (i, j), intrastrand $(i, i+1)$ and intrastrand $(i, i+2)$ neighbors. They further divided sheet residues into internal (surrounded on four sides by sheet residues) and peripheral. A χ^2 analysis was used to compare the observed distribution with that expected for random pairing. My results on three-class recognition are qualitatively similar. However, I find more recognition for the $(i, i+2)$ neighbors, and less for the (i, j) neighbors, except for the polar-polar pairs. They claim that the distribution of $(i, i+2)$ pairs does not differ significantly from a random one.

Lifson and Sander analyzed the pairwise occurrence of residues in proteins, in a specific topological relationship in the beta sheet [Lifson and Sander, 1980]. The residues are in close proximity and therefore are in a position to have energetic interactions which can contribute to the overall stability of the molecule. Lifson and Sander counted the number of occurrences of each pair, and compared the counts to those expected based on the assumption of random pairing. They found significant "recognition", or non-randomness, using a χ^2 test. They reported the likelihood ratio of observed to expected counts, along with the variance (computed as $1/\sqrt{\text{expected counts}}$).

Lifson and Sander performed this analysis separately for parallel and antiparallel beta strand arrangements. They also found that there was significant "specific recognition" over and above "nonspecific recognition" (in which the 20 amino acids are grouped into three classes: polar, neutral, and hydrophobic).

There have also been analyses of interactions at a finer level of detail than the whole side chain. Warme and Morgan [Warme and Morgan, 1978a] surveyed interactions between side-chains and 15 types of side-chain atoms in 21 proteins. They found 35 residue-atom pairs that exhibit frequencies of interaction that differ by at at least 50% from the expected values. They also tabulated residue-residue interactions, where each interaction is defined as a contacting atom pair (so one pair of residues could contribute more than one count).

Warme and Morgan [Warme and Morgan, 1978b] also surveyed atomic interactions in 21 proteins. They divided the atoms into 19 types and tabulated the atomic contacts (those whose centers were within a distance of each other that is no more than 1 angstrom plus the sum of their van der Waals' radii). By comparing observed to

expected counts, they pulled out favorable and unfavorable interactions (for example, "sulfur atoms are attracted to other sulfur atoms and avoid negatively charged oxygen atoms.").

Narayana and Argos [Narayana and Argos, 1984], like Warme and Morgan, examined the preferential association of amino acid side groups with specific side chain atoms in 44 protein structures. They used these numbers as potentials to detect structural homology in proteins with little sequence homology. They claim that the statistics show that side chains have a bias toward contact with other side chain atoms on their N-terminal side, which they say has implications for folding.

A number of other analyses of pairwise interactions are described below in the section on threading pseudopotentials.

A.2 Threading

In this section I summarize some of the related work on threading. Threading is the alignment of a sequence to a structure. A number of researchers have used this approach to the inverse folding problem in recent years. Several reviews have been published about inverse folding [Wodak and Rooman, 1993, Blundell and Johnson, 1993, Fetrow and Bryant, 1993]. Each group uses their own structure representation, pseudopotential generation, protein set, threading algorithm, and tests of the method. Of particular interest here are the structure representations, and I will describe those in more detail below.

The algorithms for threading can be straight dynamic programming provided the structure representation includes single-residue terms only [Luthy *et al.*, 1992]. When pairwise and higher terms are incorporated, heuristic algorithms are generally used [Godzik and Skolnick, 1992, Jones *et al.*, 1992]. Lathrop has developed a fast optimal algorithm that can handle higher-order terms [Lathrop and Smith, 1994].

A.2.1 Single-residue potential functions

In their profile method, Luthy et al [Luthy *et al.*, 1992, Luthy *et al.*, 1991, Luthy *et al.*, 1994] have single-residue potentials that consider solvent exposure, polar atoms, and secondary structure. They define 18 singleton structure environments. There are three secondary structure categories (alpha, beta, other). There are six categories related to the environment's polarity, which consider both the area of the sidechain buried and the fraction of surrounding atoms which are polar. The score is a straightforward log likelihood ratio score. Dynamic programming is used, with gap penalties, to align a sequence to a linear structure description.

A.2.2 Pseudo-singleton potential functions

Ouzonis and colleagues consider a set of structure representations of different complexities [Ouzounis *et al.*, 1993]. All of them involve the interaction of a residue position with its environment. The environment types are as follows:

- Two types. Inside and outside of the protein
- Five types. Contact can be made with a helix, strand, turn, coil, or solvent.
- 29 types. This includes the secondary structure state of both contacting partners and a rough idea of the topological relationship of the two residues (for example, same strand, adjacent strand, or adjacent in sequence).

Each score is weighted by the amount of contact between the residue and its neighbor in the original protein.

I termed this type of threading potential pseudo-singleton in Chapter 4. It involves the association between a residue's amino acid type and the neighbors' structure type. Optimal threading using this potential can be done using the fast dynamic programming algorithm.

A.2.3 Pairwise potential functions

Miyazawa and Jernigan computed contact potentials based on the number of occurrences of nearby pairs of residues in proteins [Miyazawa and Jernigan, 1985].

Sippl also derives pairwise potentials, but separates pairs according to the distance between residues [Sippl, 1990, Sippl and Weitckus, 1992, Casari and Sippl, 1992]. Separation of pairs into classes determined by sequence separation is also considered [Hendlich *et al.*, 1990]. Jones uses a similar technique, but adds a singleton solvation potential to express the preference of each amino acid type for the inside or outside of the protein [Jones *et al.*, 1992].

Overington and Blundell have 64 structure categories to describe each residue position in their structure representation, and they further record these values for pairs of amino acids [Johnson *et al.*, 1993, Overington *et al.*, 1992]. This is a product of two solvent accessibility terms, eight hydrogen-bonding ability categories (themselves the cross-product of three binary categories), and four secondary structure categories.

Lathrop, Smith and colleagues use singleton and pairwise potentials [Lathrop *et al.*,]. Each residue is classified according to its solvent exposure (various numbers of classes have been tried) and its secondary structure (alpha, beta, or coil). Pairs are determined by the distance between beta carbons.

Godzik and Skolnick have pairwise potentials, as well as some triplet terms [Godzik and Skolnick, 1992]. They have two structure categories, buried and exposed, in their singleton terms. They also consider all pairwise and 90 statistically significant tertiary interactions between residues that are neighbors in the protein structure. These higher-order interactions are not divided into further structural categories; a pair or triplet is defined only by proximity of the sidechains.

Bryant and Lawrence [Bryant and Lawrence, 1993] used loglinear models to analyze pairwise counts in proteins. They used the fitted parameters of the loglinear models as potentials for threading. In their paired-residue structure representation, they do not distinguish between solvent exposures or secondary structures of the residues, but they do have separate categories for different pair distances. They considered the peptide bond as an additional residue type, and distinguished among a

set of distances between side chains. They weighted the counts so that the overall contribution to the contingency table from any two proteins was the same.

Crippen investigated various sequence representations, obtained by grouping the 20 amino acid types in various ways, in deriving contact potentials [Crippen, 1991]. His goal was to reduce the representational complexity to make up for the small sample size available to estimate the potentials. He also separated contacting residue pairs based on their sequence separation.

Appendix B

Data Sets

B.1 DSSP and HSSP data bases

The DSSP and HSSP files are available by public ftp from the European Molecular Biology Laboratory. To retrieve them, ftp to ftp-heidelberg.de, and log on as anonymous. The files are located in the directory pub/databases/protein_extras/dssp and pub/databases/protein_extras/hssp.

DSSP files contain information about each residue's secondary structure and solvent exposure. HSSP files contain alignments of similar sequences.

B.2 Protein sets

B.2.1 Jones 1992 data set

These proteins are nonhomologous and well-refined.

```
2fb4l 2fb4h 351c  256ba 2aat  1abp  5acn  8adh  3adk  8atca 8atcb 2azaa 3blm
1bp2  2ca2  7cata 1cc5  1ccr  2ccya 1cd4  2cdv  3cla  2cna  4cpai 5cpa  2cpp
4cpv  1crn  2cro  1csee 1csei 1ctf  1cy3  2cyp  3dfr  4dfra 1dhfa 1eca  2er7e
1fd2  1fx1  3fxc  4fxn  3gapa 2gbp  1gcr  1gd1o 3grs  3hhba 1hip  2hlaa 2hlab
1hoe  1i1b  3icb  3icd  1l01  2lbp  6ldh  1lh1  1lrd3 2ltna 1lz1  1mba  1mbd
4mdha 2mhr  2ovo  2paba 9pap  1paz  1pcy  1pfka 3pgk  3pgm  1phh  5pti  4ptp
1rhd  2rhe  2rnt  7rsa  4rxn  2sga  4sgbi 1sn3  2sns  2sodo 2ssi  2stv  1tgsi
2tmne 4tnc  1tnfa 1ubq  1utg  9wgaa 2wrpr 1wsya 1wsyb 4xiaa 1ypia
```

B.2.2 Pdb_select.aug_1993

I used a subset of 252 proteins from the pdb_select.aug_1993 proteins from EMBL for which hssp files exist [Hobohm, Scharf, Schneider, and Sander, "Selection of representative protein data sets," Protein Science 1:409-417, 1992]. The proteins used were

```
102l  1aaf  1aaib 1aaj  1aak  1aapa 1abg  1abh  1abk  1ada  1ads  1apo  1aps
1arb  1asoa 1atna 1atx  1avha 1ayh  1baa  1bbha 1bbl  1bbo  1bbpa 1bbt1 1bbt2
```

```
1bmv1 1bmv2 1bop  1bova 1brd  1btc  1bw4  1c5a  1caj  1cas  1cbn  1cbp  1cbx
1ccr  1cd8  1cdta 1cid  1clm  1cmba 1cox  1cpca 1cpcl 1cpl  1csei 1ctaa 1d66a
1dfna 1dhr  1dpi  1dri  1eaf  1ech  1eco  1end  1epj  1erp  1etu  1ezm  1fas
1fbaa 1fc1a 1fc2c 1fdd  1fha  1fiab 1fnr  1fxia 1gky  1glag 1gly  1gmfa 1gox
1gp1a 1gpb  1gpr  1gps  1grca 1grda 1gsgp 1gsta 1hc6  1hddc 1hfi  1hgeb 1higa
1hila 1hlha 1hsba 1hsda 1ifa  1isua 1izbb 1lap  1lig  1lpe  1ltsa 1ltsc 1ltsd
1lz3  1mamh 1mbd  1mdc  1mina 1minb 1mli  1mona 1mrm  1mrt  1ms2a 1mup  1nipb
1nrca 1nxb  1ofv  1omf  1omp  1ovaa 1ovb  1pafa 1pba  1pbxa 1pcda 1pda  1pdc
1pde  1pfka 1pgd  1phg  1phh  1phs  1phy  1plc  1ppba 1ppfe 1ppl  1ppn  1ppt
1prcc 1prcm 1pte  1pyab 1pyp  1r094 1r1a2 1r1ee 1rbp  1rcb  1rea  1rhd  1rnd
1rpra 1rvea 1s01  1sas  1sdha 1sgt  1shaa 1snc  1spa  1tabi 1ten  1tfg  1tgsi
1tho  1tie  1tlk  1tmd  1tnfa 1tpt  1trb  1troa 1ttb1 1ula  1utg  1vaab 1vsga
1wsya 1wsyb 256ba 2aaa  2achb 2at2c 2avia 2azaa 2bds  2bpa1 2bpa2 2bpa3 2cbh
2ccya 2cdv  2cmd  2crd  2cro  2cts  2cyp  2dnja 2gb1  2glsa 2had  2hhrc 2hipa
2hvp  2ila  2lbp  2ltna 2ltnb 2madl 2mev1 2mev4 2mhr  2mhu  2msba 2pf2  2pia
2plv1 2plv3 2pmga 2por  2ren  2rn2  2scpa 2sga  2sici 2sn3  2snv  2stv  2tbva
2tmvp 2ztaa 3adk  3b5c  3cbh  3cd4a 3chy  3cla  3dfr  3gapa 3gbp  3grs  3il8
3pgk  3rubs 3sc2a 3sc2b 3sgbi 3sodo 3tgl  451c  4blma 4bp2  4cpai 4enl  4fgf
4fxn  4gcr  4gpd1 4icd  4rcrh 4rxn  4sbva 4sgbi 4tgf  4tms  4ts1a 5fbpa 5hir
5nn9  5p21  7apib 7tima 7xia  7znf  8abp  8acn  8adh  8atca 8atcb 8cata 8i1b
9rnt  9rubb 9wgaa 1ixa  1fbfa 1ctf  1bds  1cbh  2tgf
```

B.2.3 Rost and Sander data set

Rost and Sander cite a set of proteins, listed below. These proteins have no more than 25% sequence identity, and have resolution less than or equal to 2.5 Angstroms. Protein 1sdh is no longer in the database, so I have replaced it by 3sdh, though I have had to use the 1sdh HSSP file, because there is not one available for 3sdh. Some proteins have multiple positions cited for some atoms. I take the first of these positions as the atomic position. I ignore any reported acetyl groups at the N-terminus of the proteins, and the OXT atoms at the C-terminus.

```
256ba 9apib 7cata 6cpa  3ebx  4fxn  6hir  1l58  2mev4 1pyp  3rnt
2stv  2utga 2aat  1azu  1cbh  6cpp  5er2e 3gapa 3hmga 1lap  2or1l
1r092 7rsa  2tgpi 9wgaa 8abp  3b5c  1cc5  4cpv  1etu  2gbp  3hmgb
5ldh  1ovoa 2mhu  2rspa 1tgsi 2wrpr 6acn  1bbpa 2ccya 1crn
1fc2c 2gcr  2hmza 2lh4  2paba 1mrt  4rxn  3tima 1wsya 1acx  1bds
1cd4  1csei 1fdlh 1gd1o 5hvpa 2lhb  1paz  1ppt  1s01  6tmne 1wsyb
8adh  1bmvl 1cdta 6cts  1fdx  2glsa 2i1b  1lrd3 9pap  1rbp  3sdha
2tmvp 4xiaa 3ait  1bmv2 3cla  2cyp  1fkf  2gn5  3icb  2ltna 2pcy
1rhd  4sgbi 1tnfa 1prcc 1ak3a 3blm  3cln  5cytr 2fnr  1gp1a 7icd
2ltnb 4pfk  4rhv1 1sh1  4ts1a 1prch 2alp  4bp2  4cms  1eca  2fxb
4gr1  1il8a 5lyz  3pgm  4rhv3 2sns  2tsca 1prcl 9apia 2cab  4cpai
6dfr  1fxia 1hip  9insb 1mcpl 2phh  4rhv4 2sodb 1ubq  1prcm
```

B.2.4 Set of 55 nonhomologous, monomeric proteins

The 55 proteins listed in Tables B.1 and B.2 are a subset of the 57-protein list generated by Nambudripad and colleagues at Boston University [Nambudripad *et al.*,]. Only non-homologous, monomeric, single-domain proteins solved to high resolution by X-ray crystallography were included. Smaller proteins with very little secondary structure were excluded. The proteins were selected from Release 66 of the Brookhaven protein databank. The list consists of proteins ranging in size from 74 to 405 residues. I eliminate two additional proteins from the list, 1nar and 1ifc. 1nar does not have an HSSP file in the Sander and Schneider database. The secondary structure for 1ifc is mislabeled in the Kabsch and Sander DSSP file.

B.3 Maximum solvent accessibilities

```
A = 124
C = 94
D = 154
E = 187
F = 221
G = 89
H = 201
I = 193
K = 214
L = 199
M = 216
N = 161
P = 149
Q = 192
R = 244
S = 113
T = 151
V = 169
W = 264
Y = 237
```

B.4 Atomic Radii

```
C  1.8
H  0.8
O  1.6
N  1.6
S  1.9
F  0.65
```

Protein	PDB code	Length
ubiquitin conjugating enzyme	1AAK	150
glutaredoxin	1ABA	87
apolipophorin III	1AEP	153
alpha-lactalbumin	1ALC	122
pokeweed antiviral protein	1APA	261
endochitinase	1BAA	243
granulocyte colony-stimulating factor	1BGC	158
phospholipase	1BP2	123
glucanohydrolase	1BYH	214
phosphoribosylglycinamide formyltransferase	1CDE	210
cystatin	1CEW	108
dihydropteridine reductase	1DHR	236
endonuclease V	1END	137
phosphocarrier III	1F3G	150
alpha-amylase inhibitor	1HOE	74
lectin	1LEC	243
lysin	1LIS	131
methionine aminopeptidase	1MAT	263
myoglobin	1MBD	153
ribosomal protein S5 (prokaryotic)	1PKP	145
plastocyanin	1PLC	99
interleukin 4	1RCB	129
recoverin	1REC	185
subtilisin	1S01	275
erythrina trypsin inhibitor	1TIE	166
ubiquitin	1UBQ	76
FK-506 binding protein	1YAT	113
actinidin (sulfhydryl proteinase)	2ACT	218
carbonic anhydrase II	2CA2	256
cyclophilin A	2CPL	164
cytochrome P450CAM	2CPP	405
cytochrome C peroxidase	2CYP	293
haloalkane dehalogenase	2HAD	310
histidine-containing phosphocarrier protein	2HPR	87
lysozyme	2LZM	164
macromomycin	2MCM	112
myohemerythrin	2MHR	118
staphylococcal nuclease	2SNS	141
cytochrome C551	351C	82
adenylate kinase	3ADK	194

Table B.1: Proteins used in threading experiments.

Protein	PDB code	Length
signal transduction protein CHE*Y	3CHY	128
native elastase	3EST	240
flavodoxin	3FXN	138
basic fibroblast growth factor	4FGF	124
pepsin	4PEP	326
triacylglycerol acylhydrolase	4TGL	265
carboxypeptidase A	5CPA	307
calcium-binding parvalbumin B	5CPV	108
cytochrome C	5CYT	103
ferredoxin	5FD1	106
thermolysin	5TMN	318
ribonuclease A	7RSA	124
dihydrofolate reductase	8DFR	186
antitrypsin	9API	376
ribonuclease T1	9RNT	104

Table B.2: Proteins used in threading experiments, continued.

Appendix C

Neural Network Results

The tables in this appendix show the prediction performance on train and test sets of the neural networks in each experiment and cross-validation group. Each individual three by three table shows predicted structure down the side and target (correct) structure across the top. Also shown, to the right of the small table, is the percent correct overall. Finally, the cross-correlation coefficients are shown for alpha, beta and coil structures.

The tables are followed by ten figures that show the learning curves for each of the ten neural network experiments. In each graph, the solid lines are the test results and the dashed lines are the test results.

Experiment 1: B

Group		Train Counts			Train % corr.		Test Counts			Test % corr.	Test CC
		α	β	–			α	β	–		
1	α	4137	1066	1524	0.6	α	592	171	186	0.6	0.43
	β	507	1653	624		β	69	211	84		0.34
	–	2436	1773	7939		–	311	240	1067		0.45
2	α	4353	1203	1659	0.6	α	483	113	119	0.6	0.45
	β	446	1627	586		β	41	108	35		0.33
	–	2497	1918	8356		–	232	145	669		0.47
3	α	4427	1217	1652	0.6	α	548	73	230	0.6	0.43
	β	557	1853	735		β	75	154	97		0.31
	–	2177	1622	7829		–	268	195	881		0.38
4	α	4395	1162	1687	0.6	α	468	86	224	0.6	0.40
	β	672	1952	769		β	128	176	179		0.28
	–	2169	1607	7638		–	220	131	927		0.41
5	α	4762	1226	1729	0.6	α	228	177	187	0.6	0.31
	β	459	1541	601		β	12	168	29		0.37
	–	2444	1748	8132		–	147	254	746		0.37
6	α	4386	1103	1646	0.6	α	421	137	150	0.6	0.35
	β	564	1793	719		β	71	181	73		0.33
	–	2310	1716	8033		–	300	184	803		0.41
7	α	4282	1094	1556	0.6	α	451	119	178	0.6	0.35
	β	525	1685	656		β	84	217	69		0.36
	–	2370	1764	8148		–	340	235	817		0.37
8	α	4231	1092	1539	0.6	α	587	176	244	0.6	0.39
	β	531	1628	638		β	44	227	69		0.37
	–	2340	1708	8011		–	319	283	923		0.38
9	α	4462	1176	1651	0.6	α	327	81	128	0.6	0.40
	β	570	1859	703		β	47	92	55		0.29
	–	2475	1801	8290		–	171	105	597		0.43
10	α	4298	1149	1659	0.6	α	626	207	291	0.6	0.37
	β	449	1546	559		β	90	255	118		0.35
	–	2237	1744	7548		–	352	213	1249		0.43

Results on Experiment 1.

Experiment 2: BA

Group		Train Counts			Train % corr.		Test Counts			Test % corr.	Test CC
		α	β	–			α	β	–		
1	α	3338	457	836	63.7	α	492	77	126	63.6	0.45
	β	1131	2443	1233		β	138	309	149		0.38
	–	2611	1592	8018		–	342	236	1062		0.43
2	α	4792	1295	1802	64.0	α	518	114	128	65.8	0.48
	β	236	1326	417		β	21	88	22		0.33
	–	2268	2127	8382		–	217	164	673		0.47
3	α	4903	1281	1981	64.8	α	593	69	265	63.6	0.46
	β	532	2025	862		β	62	181	113		0.37
	–	1726	1386	7373		–	236	172	830		0.38
4	α	4587	941	1650	65.2	α	486	83	227	62.4	0.42
	β	708	2356	1019		β	127	200	205		0.31
	–	1941	1424	7425		–	203	110	898		0.42
5	α	5116	1344	1799	64.1	α	247	200	179	56.8	0.34
	β	166	1026	288		β	3	97	21		0.28
	–	2383	2145	8375		–	137	302	762		0.36
6	α	4238	856	1318	65.0	α	415	94	119	61.9	0.41
	β	532	1822	656		β	56	185	71		0.36
	–	2490	1934	8424		–	321	223	836		0.40
7	α	4848	1142	1922	65.2	α	501	140	196	60.1	0.37
	β	514	1926	823		β	99	234	94		0.35
	–	1815	1475	7615		–	275	197	774		0.40
8	α	4462	1009	1619	64.9	α	635	159	239	62.7	0.45
	β	564	1866	796		β	47	269	99		0.39
	–	2076	1553	7773		–	268	258	898		0.40
9	α	4669	1042	1624	64.8	α	348	86	118	64.8	0.44
	β	509	1910	696		β	38	85	57		0.28
	–	2329	1884	8324		–	159	107	605		0.45
10	α	4713	1271	1740	63.8	α	667	239	325	62.5	0.37
	β	196	1119	332		β	46	177	51		0.33
	–	2075	2049	7694		–	355	259	1282		0.42

Results on Experiment 2.

Experiment 3: P

Group		Train Counts			Train % corr.		Test Counts			Test % corr.	Test CC
		α	β	–			α	β	–		
1	α	1742	294	481	56.4	α	275	57	61	58.7	0.31
	β	1937	2538	1663		β	274	352	182		0.34
	–	3401	1660	7943		–	423	213	1094		0.42
2	α	3961	1412	1795	59.8	α	450	131	115	62.3	0.39
	β	328	1159	382		β	26	75	21		0.28
	–	3007	2177	8424		–	280	160	687		0.44
3	α	4355	1534	2176	60.3	α	559	80	279	61.9	0.40
	β	539	1582	661		β	75	152	80		0.33
	–	2267	1576	7379		–	257	190	849		0.36
4	α	3707	1060	1628	59.7	α	402	59	202	60.5	0.36
	β	1047	2128	1138		β	173	198	191		0.29
	–	2482	1533	7328		–	241	136	937		0.39
5	α	4403	1634	1917	59.4	α	217	212	185	53.8	0.26
	β	169	708	205		β	2	66	12		0.23
	–	3093	2173	8340		–	168	321	765		0.31
6	α	2746	720	891	59.1	α	307	79	78	59.0	0.34
	β	688	1485	574		β	46	173	59		0.36
	–	3826	2407	8933		–	439	250	889		0.36
7	α	4933	1773	2848	59.8	α	530	191	303	56.5	0.29
	β	430	1390	626		β	78	191	64		0.32
	–	1814	1380	6886		–	267	189	697		0.34
8	α	4171	1296	1936	61.1	α	551	204	292	57.2	0.31
	β	568	1514	678		β	56	221	73		0.34
	–	2363	1618	7574		–	343	261	871		0.33
9	α	3532	1050	1382	59.7	α	267	69	86	62.6	0.37
	β	602	1539	607		β	34	86	43		0.31
	–	3373	2247	8655		–	244	123	651		0.40
10	α	4524	1730	2254	59.5	α	641	291	384	59.0	0.30
	β	211	832	264		β	42	150	59		0.28
	–	2249	1877	7248		–	385	234	1215		0.38

Results on Experiment 3.

Experiment 4: PA

Group		Train Counts			Train % corr.		Test Counts			Test % corr.	Test CC
		α	β	–			α	β	–		
1	α	2226	258	500	58.6	α	349	59	85	61.3	0.36
	β	1761	2625	1742		β	230	368	173		0.39
	–	3093	1609	7845		–	393	195	1079		0.44
2	α	1928	519	894	62.6	α	481	127	125	63.9	0.43
	β	139	597	243		β	20	84	21		0.32
	–	999	920	3700		–	255	155	677		0.45
3	α	4596	1488	2244	61.3	α	597	67	277	63.1	0.45
	β	479	1648	697		β	50	162	99		0.36
	–	2086	1556	7275		–	244	193	832		0.36
4	α	4139	955	1716	61.6	α	449	73	215	61.0	0.39
	β	948	2383	1315		β	157	197	213		0.29
	–	2149	1383	7063		–	210	123	902		0.40
5	α	4966	1746	2293	60.4	α	245	226	205	53.7	0.30
	β	125	733	194		β	0	61	17		0.21
	–	2574	2036	7975		–	142	312	740		0.32
6	α	2694	484	616	60.5	α	309	38	64	60.5	0.40
	β	656	1588	589		β	54	191	59		0.39
	–	3910	2540	9193		–	429	273	903		0.36
7	α	5012	1562	2749	61.0	α	541	184	283	57.6	0.32
	β	443	1598	755		β	89	208	85		0.32
	–	1722	1383	6856		–	245	179	696		0.36
8	α	4356	1283	2023	61.6	α	606	190	278	59.8	0.38
	β	536	1612	744		β	49	243	90		0.36
	–	2210	1533	7421		–	295	253	868		0.36
9	α	4128	1025	1590	61.9	α	310	79	112	63.8	0.40
	β	622	1853	812		β	31	98	53		0.35
	–	2757	1958	8242		–	204	101	615		0.42
10	α	4663	1546	2208	61.0	α	631	297	417	57.6	0.27
	β	223	1059	365		β	63	154	68		0.26
	–	2098	1834	7193		–	374	224	1173		0.36

Results on Experiment 4.

Experiment 5: H

Group		Train Counts			Train % corr.		Test Counts			Test % corr.	Test CC
		α	β	–			α	β	–		
1	α	1427	885	957	51.2	α	185	138	116	52.1	0.08
	β	736	1123	588		β	100	179	58		0.28
	–	4917	2484	8542		–	687	305	1163		0.28
2	α	1647	1107	1171	51.5	α	151	88	61	48.9	0.10
	β	607	1050	467		β	57	75	36		0.20
	–	5042	2591	8963		–	548	203	726		0.25
3	α	1558	963	1036	51.5	α	162	76	147	50.0	0.06
	β	898	1369	733		β	111	110	72		0.20
	–	4705	2360	8447		–	618	236	989		0.19
4	α	1511	971	1074	50.8	α	202	85	144	55.1	0.14
	β	991	1410	748		β	119	128	117		0.22
	–	4734	2340	8272		–	495	180	1069		0.26
5	α	2018	1332	1428	51.1	α	74	101	85	52.9	0.08
	β	552	914	393		β	17	111	32		0.25
	–	5095	2269	8641		–	296	387	845		0.23
6	α	1541	951	1074	51.5	α	154	119	107	49.7	0.06
	β	835	1258	660		β	92	136	57		0.24
	–	4884	2403	8664		–	546	247	862		0.25
7	α	1314	848	938	51.3	α	200	107	114	49.6	0.12
	β	847	1225	630		β	107	165	70		0.24
	–	5016	2470	8792		–	568	299	880		0.24
8	α	1446	870	974	51.8	α	174	145	104	48.4	0.07
	β	788	1175	594		β	89	165	82		0.22
	–	4868	2383	8620		–	687	376	1050		0.22
9	α	1444	902	989	51.2	α	112	55	61	53.7	0.13
	β	903	1357	687		β	62	77	47		0.23
	–	5160	2577	8968		–	371	146	672		0.27
10	α	1866	1240	1322	51.3	α	318	213	294	51.6	0.09
	β	555	960	409		β	103	152	78		0.21
	–	4563	2239	8035		–	647	310	1286		0.24

Results on Experiment 5.

Experiment 6: HA

Group		Train Counts			Train % corr.		Test Counts			Test % corr.	Test CC
		α	β	–			α	β	–		
1	α	1453	359	596	53.1	α	188	60	84	53.6	0.18
	β	1640	2202	1638		β	229	303	173		0.30
	–	3987	1931	7853		–	555	259	1080		0.31
2	α	3414	1499	2201	54.2	α	338	119	126	55.1	0.26
	β	417	866	401		β	31	68	32		0.23
	–	3465	2383	7999		–	387	179	665		0.31
3	α	3537	1434	2207	55.2	α	402	79	285	54.6	0.24
	β	629	1440	799		β	79	140	88		0.29
	–	2995	1818	7210		–	410	203	835		0.23
4	α	3105	1130	1912	54.1	α	399	104	256	57.5	0.29
	β	1090	1928	1275		β	132	165	178		0.26
	–	3041	1663	6907		–	285	124	896		0.34
5	α	3228	1320	1691	54.7	α	145	149	109	53.1	0.21
	β	234	647	266		β	3	59	22		0.18
	–	4203	2548	8505		–	239	391	831		0.26
6	α	2092	685	938	54.7	α	242	73	96	54.3	0.24
	β	755	1328	688		β	85	146	58		0.26
	–	4413	2599	8772		–	465	283	872		0.29
7	α	3433	1296	2265	54.9	α	405	173	216	53.7	0.23
	β	756	1433	840		β	112	201	105		0.27
	–	2988	1814	7255		–	358	197	743		0.31
8	α	3098	1263	1957	54.9	α	422	164	213	54.6	0.26
	β	722	1364	773		β	79	224	102		0.30
	–	3282	1801	7458		–	449	298	921		0.29
9	α	2738	1036	1466	54.9	α	243	73	123	58.6	0.28
	β	750	1435	737		β	38	79	40		0.29
	–	4019	2365	8441		–	264	126	617		0.33
10	α	3520	1520	2108	54.7	α	499	271	467	51.4	0.15
	β	311	735	325		β	79	117	58		0.19
	–	3153	2184	7333		–	490	287	1133		0.24

Results on Experiment 6.

Experiment 7: RP

Group		Train Counts			Train % corr.		Test Counts			Test % corr.	Test CC
		α	β	–			α	β	–		
1	α	191	47	90	46.6	α	35	5	7	46.4	0.11
	β	572	750	845		β	80	88	93		0.10
	–	6317	3695	9152		–	857	529	1237		0.09
2	α	1075	388	950	43.9	α	87	44	106	39.3	-.02
	β	1172	1449	2238		β	125	79	118		0.07
	–	5049	2911	7413		–	544	243	599		0.03
3	α	3123	1419	2298	49.8	α	380	100	235	53.6	0.23
	β	54	229	279		β	11	19	22		0.08
	–	3984	3044	7639		–	500	303	951		0.19
4	α	3597	1514	2638	49.6	α	402	130	400	49.9	0.18
	β	329	637	751		β	28	49	115		0.08
	–	3310	2570	6705		–	386	214	815		0.12
5	α	1725	576	930	49.7	α	62	101	78	48.5	0.06
	β	0	11	18		β	0	2	3		0.01
	–	5940	3928	9514		–	325	496	881		0.13
6	α	760	257	349	48.5	α	62	35	53	44.2	0.04
	β	25	58	71		β	2	6	15		0.01
	–	6475	4297	9978		–	728	461	958		0.03
7	α	5493	2836	5770	45.6	α	654	350	598	44.6	0.17
	β	17	72	81		β	2	4	5		0.02
	–	1667	1635	4509		–	219	217	461		0.14
8	α	4091	1894	3479	49.6	α	533	290	472	45.1	0.16
	β	48	160	196		β	2	18	20		0.06
	–	2963	2374	6513		–	415	378	744		0.12
9	α	1874	751	1135	49.4	α	191	50	69	56.0	0.29
	β	38	113	142		β	0	7	12		0.06
	–	5595	3972	9367		–	354	221	699		0.24
10	α	3742	1676	2973	49.7	α	569	283	572	48.5	0.16
	β	4	37	47		β	5	5	11		0.01
	–	3238	2726	6746		–	494	387	1075		0.14

Results on Experiment 7.

Experiment 8: RHA

Group		Train Counts			Train % corr.		Test Counts			Test % corr.	Test CC
		α	β	–			α	β	–		
1	α	260	64	126	46.9	α	55	10	10	47.0	0.14
	β	464	643	706		β	63	74	77		0.09
	–	6356	3785	9255		–	854	538	1250		0.10
2	α	3736	1690	3043	49.8	α	395	145	279	48.5	0.16
	β	24	81	100		β	3	9	5		0.08
	–	3536	2977	7458		–	358	212	539		0.15
3	α	3023	1339	2171	49.9	α	370	89	226	53.4	0.24
	β	68	221	269		β	10	15	21		0.06
	–	4070	3132	7776		–	511	318	961		0.18
4	α	3707	1584	2756	49.4	α	415	136	412	50.2	0.18
	β	302	592	742		β	23	49	108		0.09
	–	3227	2545	6596		–	378	208	810		0.12
5	α	1888	643	1062	49.8	α	66	108	87	48.3	0.05
	β	1	7	14		β	0	0	1		-.02
	–	5776	3865	9386		–	321	491	874		0.12
6	α	746	235	321	48.5	α	61	32	47	44.4	0.05
	β	26	51	66		β	1	6	15		0.01
	–	6488	4326	10011		–	730	464	964		0.03
7	α	5621	2950	6009	45.1	α	668	368	620	44.4	0.16
	β	19	57	66		β	2	6	4		0.05
	–	1537	1536	4285		–	205	197	440		0.14
8	α	4326	2069	3906	48.7	α	570	335	547	43.8	0.13
	β	39	117	140		β	2	13	14		0.05
	–	2737	2242	6142		–	378	338	675		0.11
9	α	1770	704	1081	49.2	α	182	47	64	55.7	0.28
	β	35	102	132		β	0	6	11		0.05
	–	5702	4030	9431		–	363	225	705		0.24
10	α	3976	1795	3308	49.2	α	608	303	647	47.4	0.15
	β	8	46	53		β	6	5	11		0.01
	–	3000	2598	6405		–	454	367	1000		0.13

Results on Experiment 8.

Experiment 9: RH

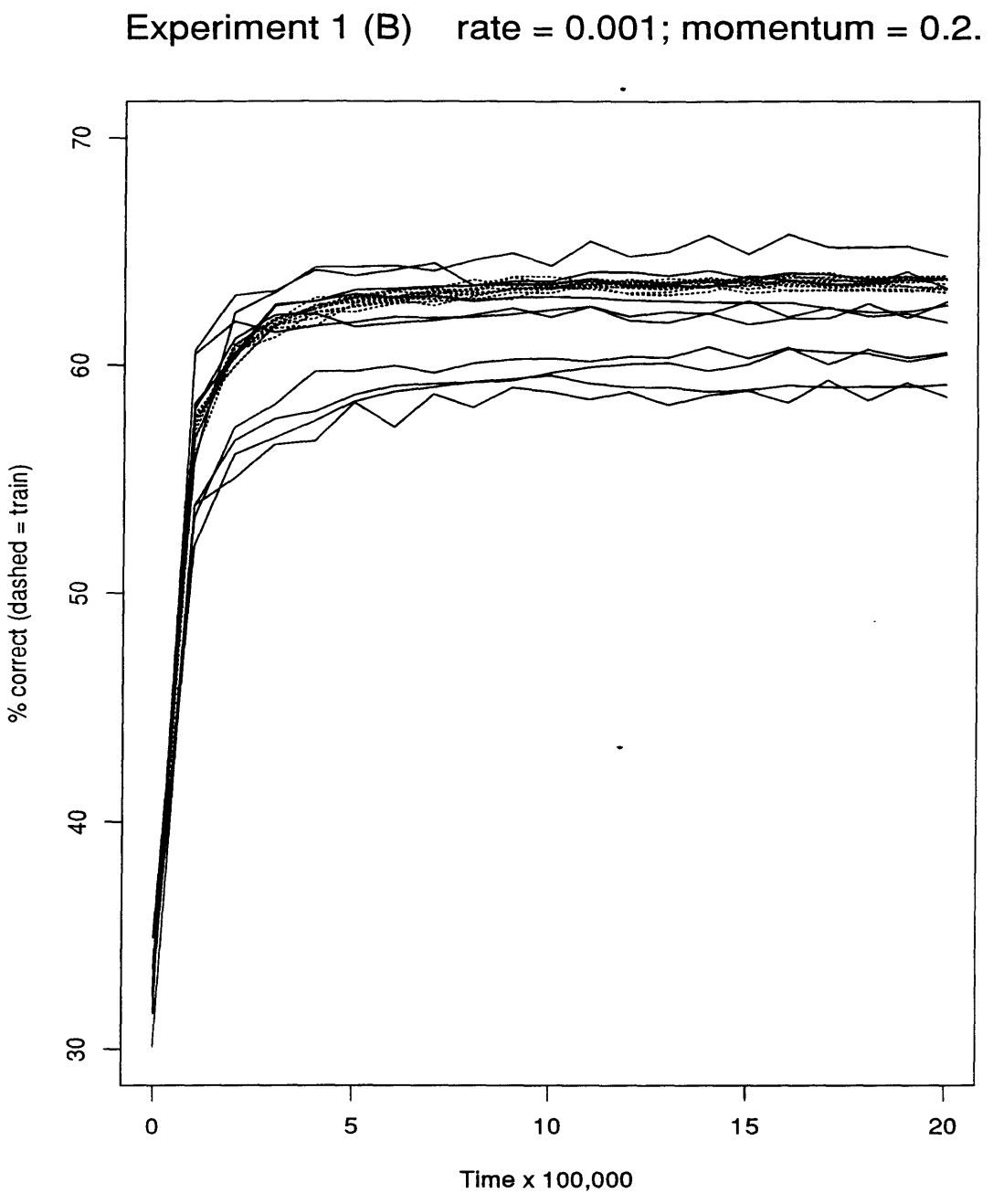
Group		Train Counts			Train % corr.		Test Counts			Test % corr.	Test CC
		α	β	–			α	β	–		
1	α	1164	289	910	47.7	α	194	37	84	49.4	0.21
	β	0	0	0		β	0	0	0		0.00
	–	5916	4203	9177		–	778	585	1253		0.13
2	α	1643	417	1275	48.4	α	187	55	181	42.6	0.06
	β	0	0	0		β	0	0	0		0.00
	–	5653	4331	9326		–	569	311	642		-.01
3	α	1610	504	1313	47.6	α	225	14	133	51.6	0.22
	β	0	0	0		β	0	0	0		0.00
	–	5551	4188	8903		–	666	408	1075		0.10
4	α	1961	585	1570	47.5	α	210	55	238	51.3	0.10
	β	0	0	0		β	0	0	0		0.00
	–	5275	4136	8524		–	606	338	1092		0.05
5	α	1836	466	1435	48.0	α	96	61	111	48.6	0.16
	β	0	0	0		β	0	0	0		0.00
	–	5829	4049	9027		–	291	538	851		0.06
6	α	1681	449	1348	48.2	α	186	53	125	46.9	0.15
	β	0	0	0		β	0	0	0		0.00
	–	5579	4163	9050		–	606	449	901		0.09
7	α	1503	384	1165	48.5	α	165	44	118	44.3	0.13
	β	0	0	0		β	0	0	0		0.00
	–	5674	4159	9195		–	710	527	946		0.05
8	α	1538	423	1227	48.3	α	185	37	108	45.7	0.18
	β	0	0	0		β	0	0	0		0.00
	–	5564	4005	8961		–	765	649	1128		0.08
9	α	1490	391	1160	47.7	α	146	26	85	52.5	0.21
	β	0	0	0		β	0	0	0		0.00
	–	6017	4445	9484		–	399	252	695		0.14
10	α	1920	522	1532	47.9	α	229	112	275	47.4	0.06
	β	0	0	0		β	0	0	0		0.00
	–	5064	3917	8234		–	839	563	1383		0.04

Results on Experiment 9.

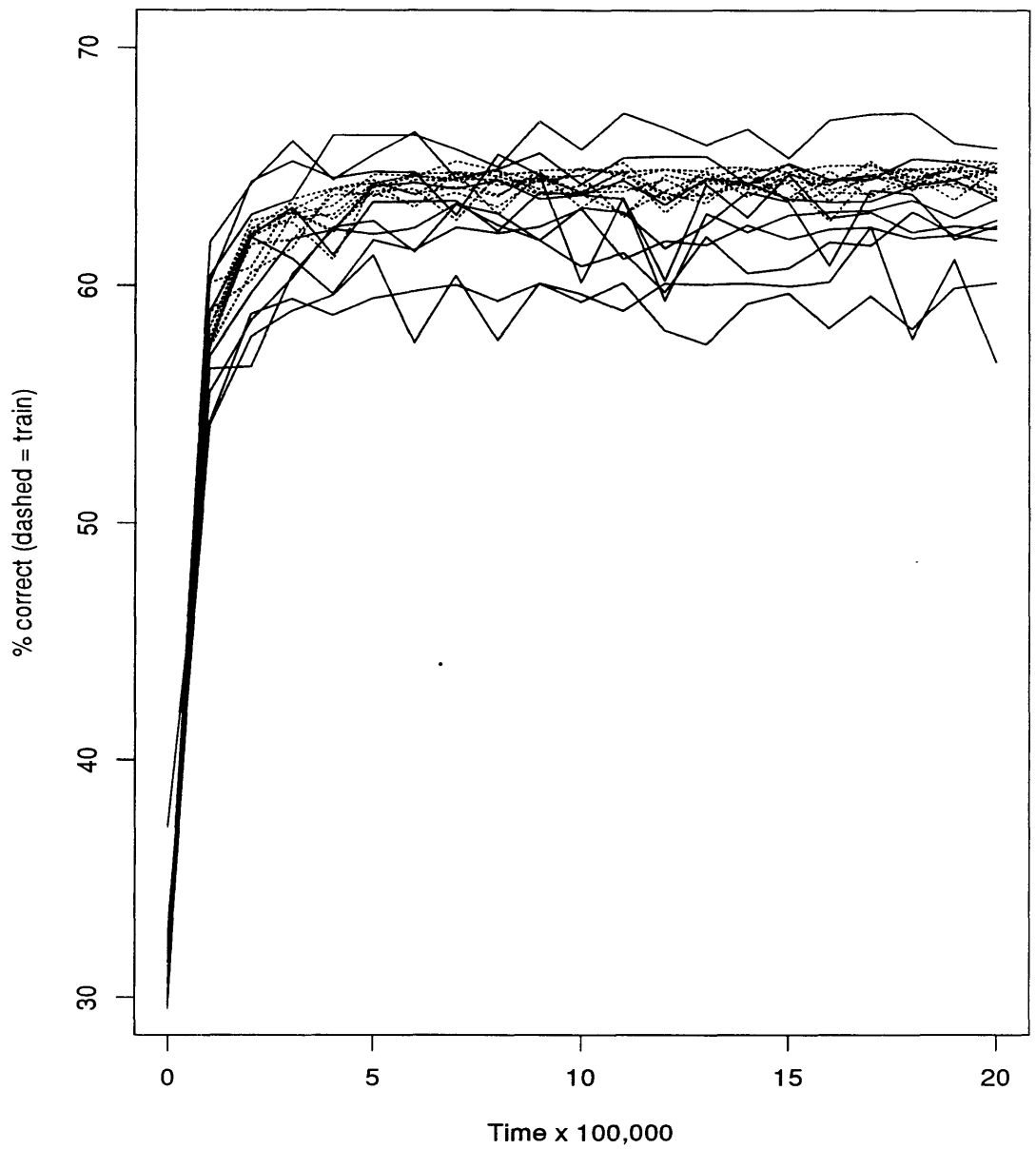
Experiment 10: RHA

Group		Train Counts			Train % corr.		Test Counts			Test % corr.	Test CC
		α	β	$-$			α	β	$-$		
1	α	458	100	309	47.3	α	70	11	20	47.3	0.15
	β	0	0	1		β	0	0	0		0.00
	$-$	6622	4392	9777		$-$	902	611	1317		0.10
2	α	1879	552	1515	48.4	α	213	61	202	42.9	0.07
	β	0	0	0		β	0	0	0		0.00
	$-$	5417	4196	9086		$-$	543	305	621		-.00
3	α	1871	664	1578	47.5	α	270	24	179	51.4	0.22
	β	13	0	24		β	0	1	3		0.01
	$-$	5277	4028	8614		$-$	621	397	1026		0.09
4	α	2373	795	1992	47.4	α	264	75	295	51.2	0.12
	β	13	4	20		β	2	0	0		-.01
	$-$	4850	3922	8082		$-$	550	318	1035		0.07
5	α	1316	312	961	47.8	α	68	29	65	49.5	0.17
	β	0	0	0		β	0	0	0		0.00
	$-$	6349	4203	9501		$-$	319	570	897		0.06
6	α	1381	346	1040	48.2	α	151	45	99	46.5	0.14
	β	0	0	0		β	0	0	0		0.00
	$-$	5879	4266	9358		$-$	641	457	927		0.08
7	α	2442	827	2131	48.3	α	264	110	234	43.6	0.10
	β	4	0	1		β	0	0	0		0.00
	$-$	4731	3716	8228		$-$	611	461	830		0.04
8	α	2088	669	1791	48.3	α	234	75	154	45.8	0.16
	β	0	0	0		β	0	0	0		0.00
	$-$	5014	3759	8397		$-$	716	611	1082		0.09
9	α	1001	231	692	47.6	α	101	18	58	51.3	0.17
	β	0	0	0		β	0	0	0		0.00
	$-$	6506	4605	9952		$-$	444	260	722		0.11
10	α	1837	479	1414	48.1	α	215	100	272	47.1	0.05
	β	0	0	0		β	0	0	0		0.00
	$-$	5147	3960	8352		$-$	853	575	1386		0.02

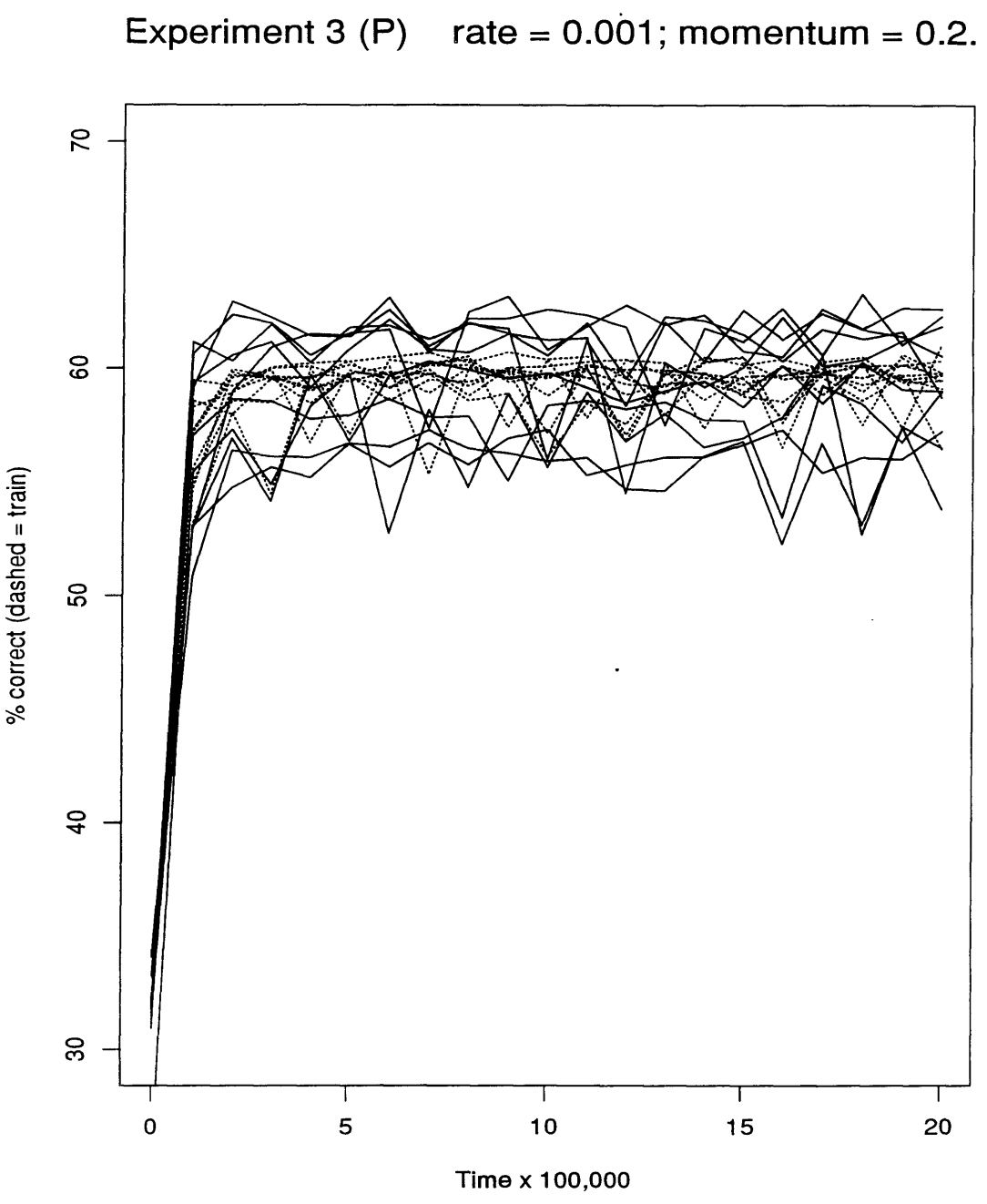
Results on Experiment 10.



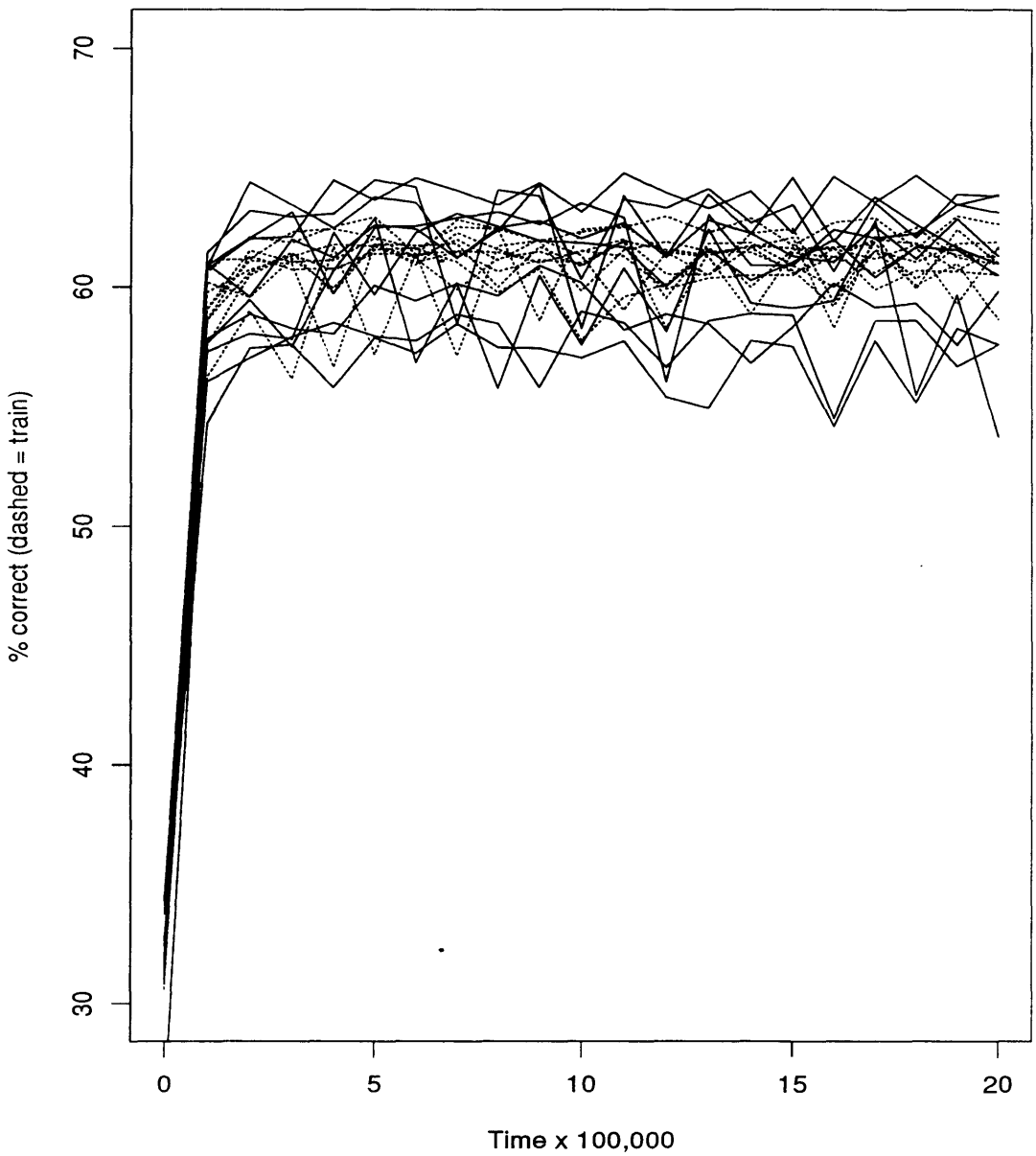
Learning Curves for neural network experiment 1.



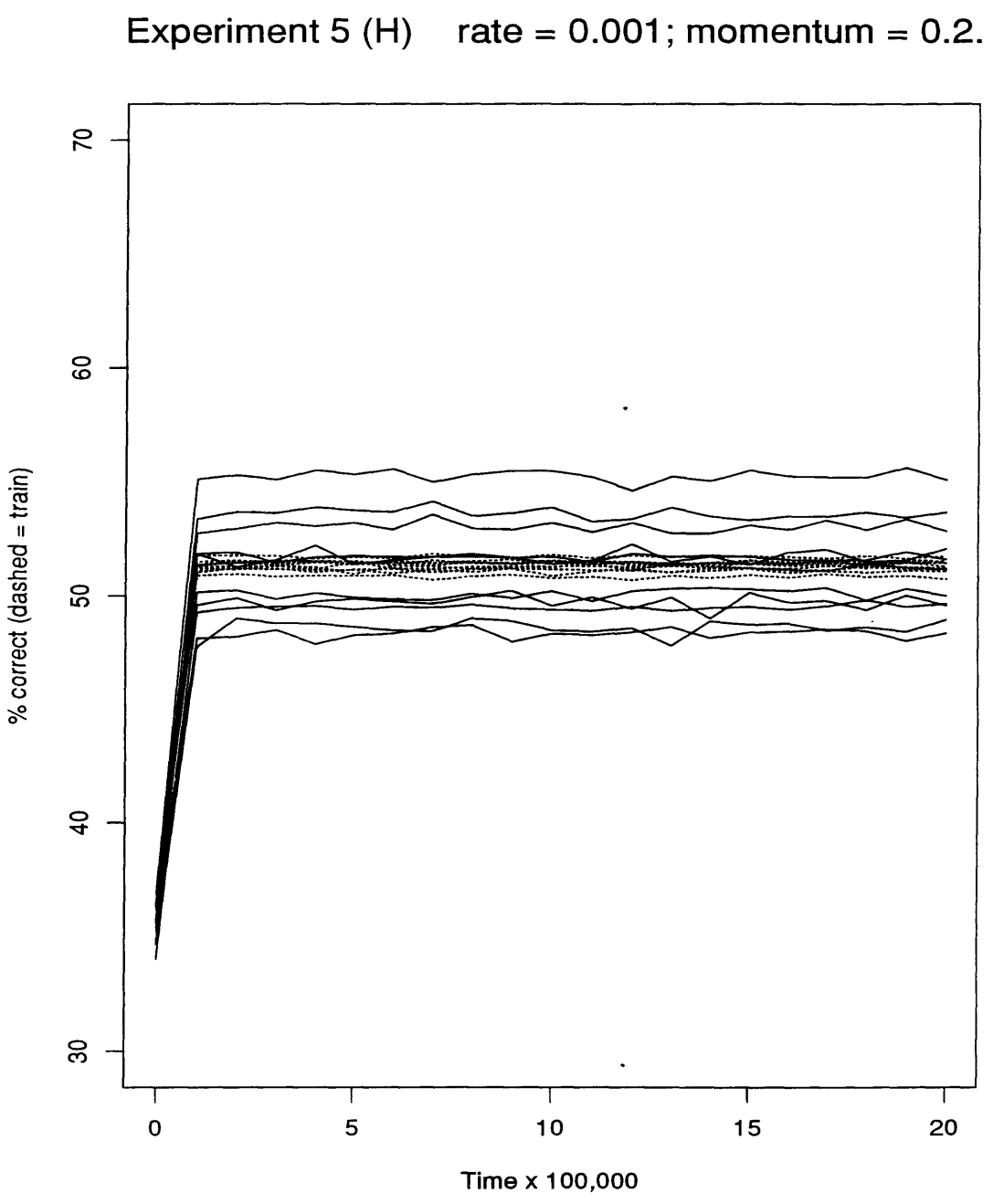
Learning Curves for neural network experiment 2.



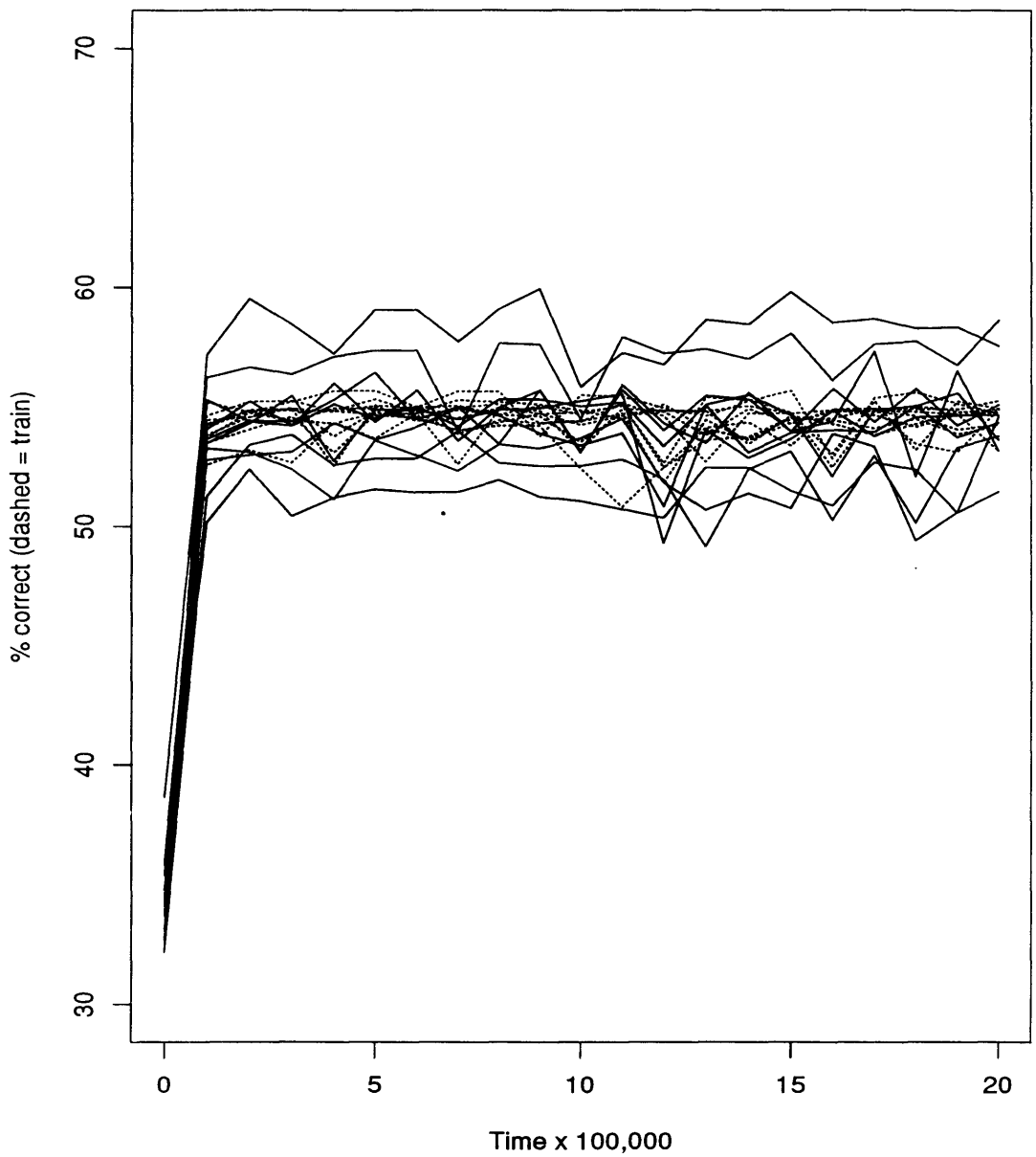
Learning Curves for neural network experiment 3.



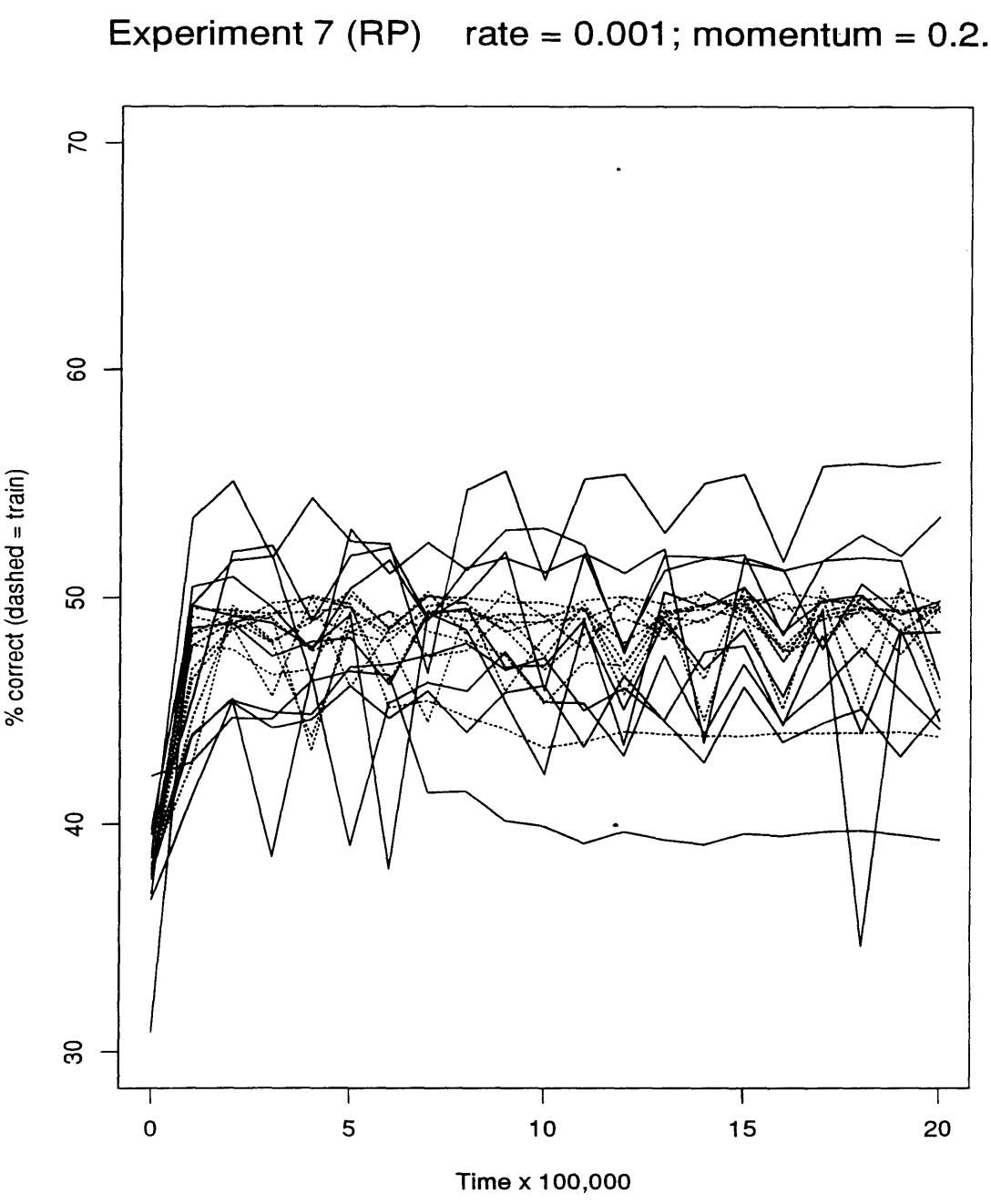
Learning Curves for neural network experiment 4.



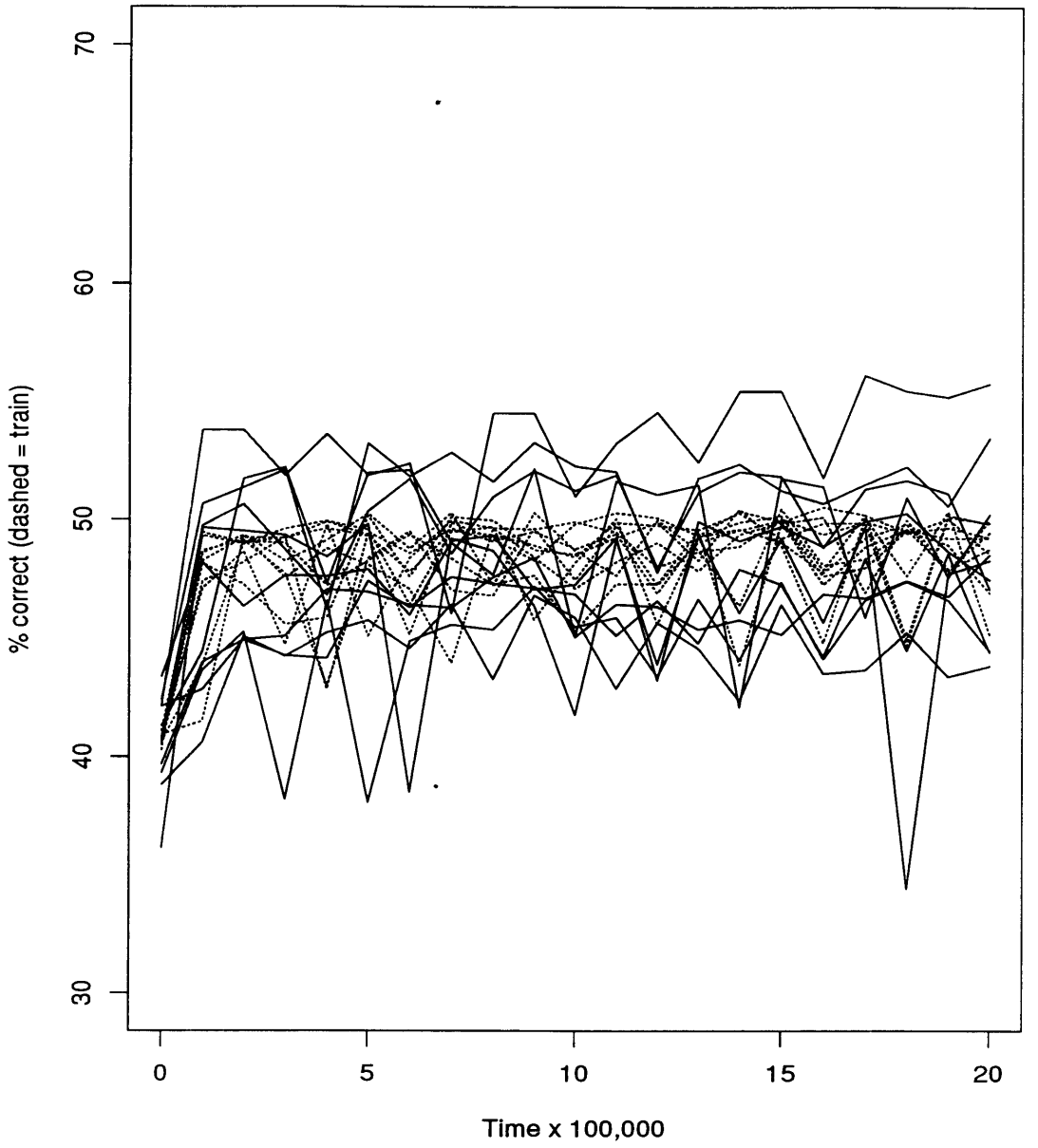
Learning Curves for neural network experiment 5.



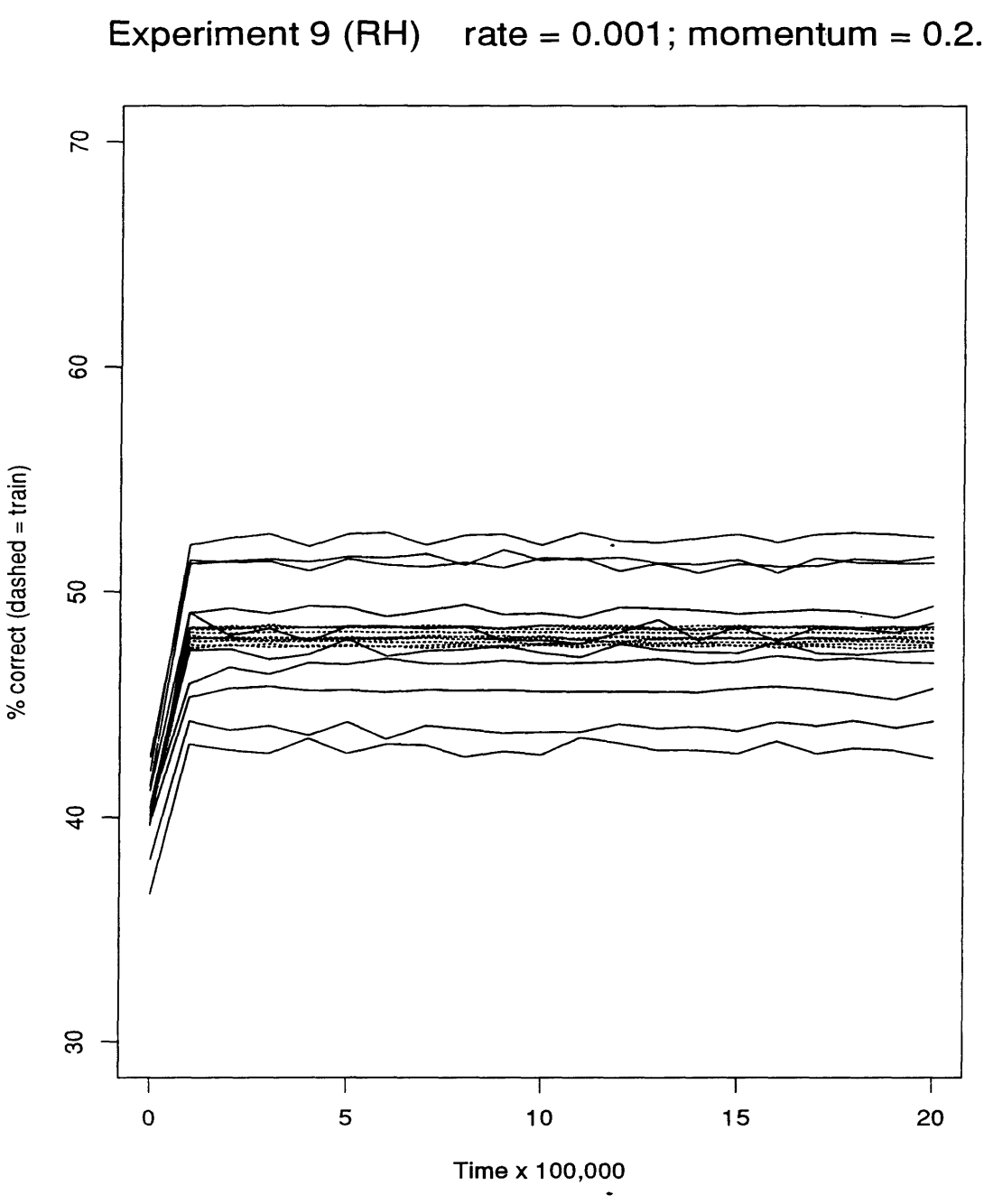
Learning Curves for neural network experiment 6.



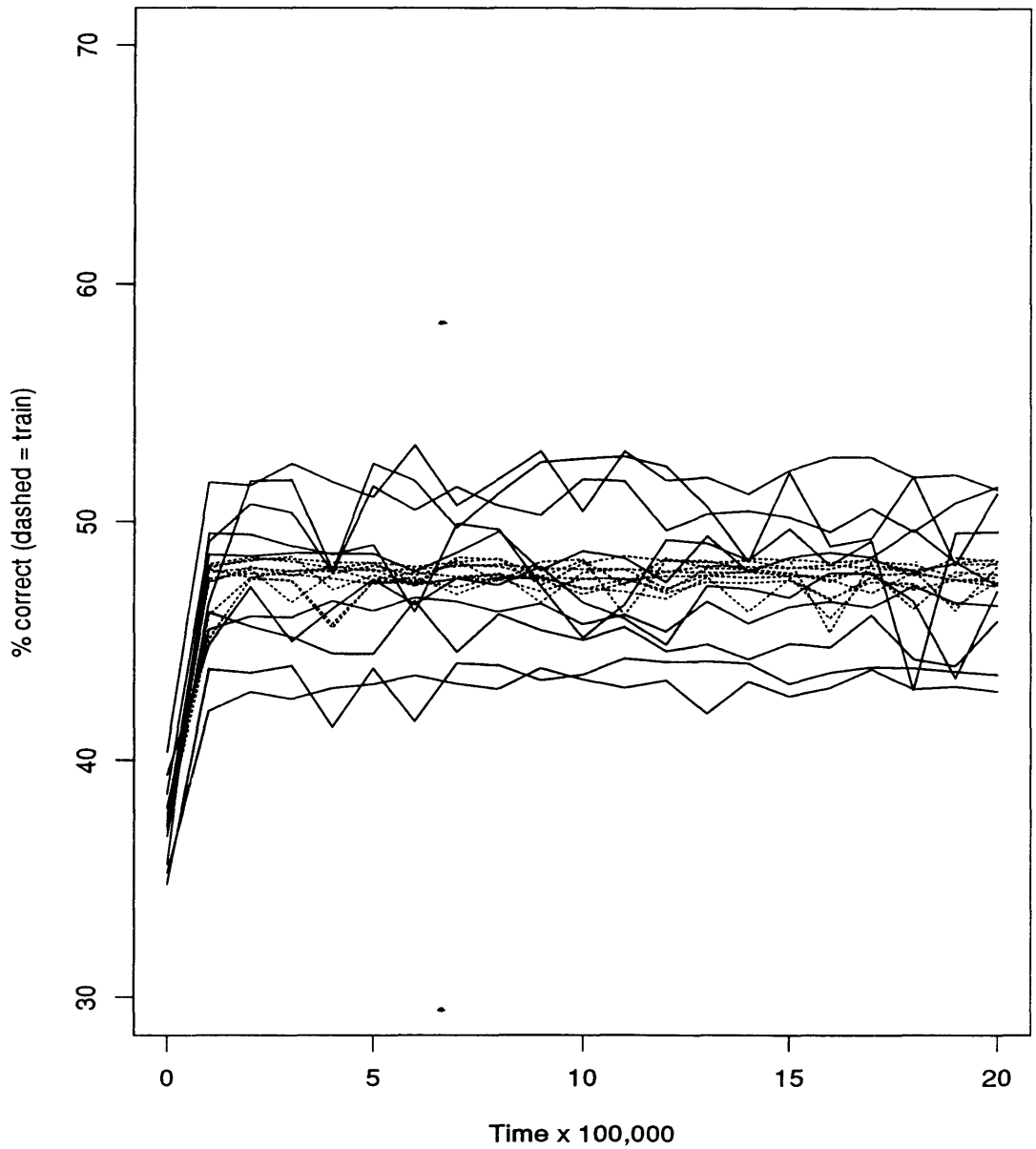
Learning Curves for neural network experiment 7.



Learning Curves for neural network experiment 8.



Learning Curves for neural network experiment 9.



Learning Curves for neural network experiment 10.

Bibliography

[Abola *et al.*, 1987] E. E. Abola, F. C. Bernstein, S. H. Bryant, T. F. Koetzle, and J. Weng. *Protein Data Bank*, pages 107–132. Data Commission fo the International Union of Crystallography, Bonn/Cambridge/Chester, 1987.

[Anfinsen *et al.*, 1961] C. B. Anfinsen, E. Haber, M. Sela, and F. H. White. The kinetics of formation of native ribonuclease during oxidation of the reduced polypeptide chain. *Proceedings of the National Academy of Sciences*, 47:1309–1314, 1961.

[Argos and Palau, 1982] Patrick Argos and Jaume Palau. Amino acid distribution in protein secondary structures. *Int. J. Peptide Res.*, 19:380–303, 1982.

[Argos, 1987] Patrick Argos. Analysis of sequence-similar pentapeptides in unrelated protein tertiary structures. *Journal of Molecular Biology*, 197:331–348, 1987.

[Baumann *et al.*, 1989] G. Baumann, C. Frommel, and C. Sander. Polarity as a criterion in protein design. *Protein Engineering*, 2(5):329–334, 1989.

[Becker *et al.*, 1988] R. A. Becker, J. M. Chambers, and A. R. Wilks. *The New S Language*. Wadsworth, Pacific Grove, California, 1988.

[Benner *et al.*, 1994] Steven Benner, Ian Badcoe, Mark Cohen, and Dietlind Gerloff. Bona fide prediction of aspects of protein conformation; assigning interior and surface residues from patterns of variation and conservation in homologous protein sequences. *Journal of Molecular Biology*, 235:926–958, 1994.

[Bernstein *et al.*, 1977] F. C. Bernstein, T. F. Koetzle, G. J. B. Williams, E. F. Meyer Jr., M. D. Brice, J. R. Rodgers, O. Kennard, T. Shimanouchi, and M. Tasumi. The protein data bank: A computer-based archival file for macromolecular structures. *Journal of Molecular Biology*, 112:535–542, 1977.

[Blundell and Johnson, 1993] Tom L. Blundell and Mark S. Johnson. Catching a common fold. *Protein Science*, 2:877–883, 1993.

[Bowie *et al.*, 1991] James U. Bowie, Roland Luthy, and David Eisenberg. A method to identify protein sequences that fold into a known three-dimensional structure. *Science*, 253:164–170, 1991.

[Bryant and Lawrence, 1993] Stephen H. Bryant and Charles E. Lawrence. An empirical energy function for threading protein sequence through the folding motif. *Proteins*, 16:92–112, 1993.

[Bycroft *et al.*, 1990] Mark Bycroft, Andreas Matouschek, James T. Kellis Jr., Luis Serrano, and Alan R. Fersht. Detection and characterization of a folding intermediate in barnase by nmr. *Nature*, 346:488–490, 1990.

[Casari and Sippl, 1992] Georg Casari and Manfred J. Sippl. Structure-derived hydrophobic potential: Hydrophobic potential derived from x-ray structures of globular proteins is able to identify native folds. *Journal of Molecular Biology*, 224:725–732, 1992.

[Chiche *et al.*, 1990] Chiche, Gregoret, Cohen, and Kollman. Protein model structure evaluation using the solvation free energy of folding. *Proceedings of the National Academy of Sciences*, 87:3240–3243, 1990.

[Chothia and Janin, 1982] Cyrus Chothia and Joel Janin. Orthogonal packing of beta-pleated sheets in proteins. *Biochemistry*, 21:3955–3965, 1982.

[Chou and Fasman, 1974] P. Y. Chou and G. D. Fasman. Conformational parameters for amino acids in helical, beta-sheet, and random coil regions calculated from proteins. *Biochemistry*, 13:222–245, 1974.

[Chou and Fasman, 1978] P. Y. Chou and G. D. Fasman. Prediction of the secondary structure of proteins from their amino acid sequence. *Advances in Enzymology*, 47:45–148, 1978.

[Cid *et al.*, 1992] Hilda Cid, Marta Bunster, Mauricio Canales, and Felipe Gazitua. Hydrophobicity and structural classes in proteins. *Protein Engineering*, 5(5):373–375, 1992.

[Cohen and Kuntz, 1987] F. E. Cohen and I. D. Kuntz. Prediction of the three-dimensional structure of human growth hormone. *Proteins*, pages 162–166, 1987.

[Cohen *et al.*, 1979] Fred E. Cohen, Timothy J. Richmond, and Frederic M. Richards. Protein folding: Evaluation of some simple rules for the assembly of helices into tertiary structures with myoglobin as an example. *J. Mol. Biol.*, 132:275–288, 1979.

[Cohen *et al.*, 1983] Fred E. Cohen, Robert M. Abarbanel, I. D. Kuntz, and Robert J. Fletterick. Secondary structure assignment for α/β proteins by a combinatorial approach. *Biochemistry*, 22:4894–4904, 1983.

[Cohen *et al.*, 1986] F. E. Cohen, R. M. Abarbanel, I. D. Kuntz, and R. J. Fletterick. Turn prediction in proteins using a pattern-matching approach. *Biochemistry*, 25:266–275, 1986.

[Collin Stultz and Smith, 1993] James White Collin Stultz and Temple Smith. Structural analysis based on state-space modeling. *Protein Science*, 2:305–314, 1993.

[Cornette *et al.*, 1987] James L. Cornette, Kemp B. Cease, Hanah Margalit, John L. Spouge, Jay A. Berzofsky, and Charles DeLisi. Hydrophobic scales and computational techniques for detecting amphipathic structures in proteins. *Journal of Molecular Biology*, 195:659–685, 1987.

[Creighton, 1978] T. E. Creighton. Experimental studies of protein folding and unfolding. *Progress in Biophysics and Molecular Biology*, 33:231–297, 1978.

[Crippen, 1991] Gordon M. Crippen. Prediction of protein folding from amino acid sequence over discrete conformation spaces. *Biochemistry*, 30:4232–4237, 1991.

[Dubchak *et al.*,] Inna Dubchak, Stephen R. Holbrook, and Sung-Hou Kim. Prediction of protein folding class from amino acid composition. University of California at Berkeley.

[Eisenberg *et al.*, 1982] D. Eisenberg, R. M. Weiss, and T. C. Terwilliger. The helical hydrophobic moment: a measure of the amphiphilicity of a helix. *Nature*, 299:371–374, 1982.

[Eisenberg *et al.*, 1984a] D. Eisenberg, E. Schwarz, M. Komaromy, and R. Wall. Analysis of membrane and surface protein sequences with the hydrophobic moment plot. *Journal of molecular biology*, 179:125–142, 1984.

[Eisenberg *et al.*, 1984b] D. Eisenberg, R. M. Weiss, and T. C. Terwilliger. The hydrophobic moment detects periodicity in protein hydrophobicity. *Proceedings of the National Academy of Science USA*, 81:140–144, 1984.

[Fasman, 1989] Gerald R. Fasman. *Prediction of Protein Structure and the Principles of Protein Conformation*, chapter The Development of the Prediction of Protein Structure, pages 193–316. Plenum Press, 233 Spring Street, New York, NY 10013, 1989.

[Feldman, 1976] R. J. Feldman. *AMSOM tlas fo Molecular Structures on Microfiche.* U.S. NIH, 1976.

[Ferran and Ferrara, 1992] Edgardo A. Ferran and Pascual Ferrara. Clustering proteins into families using artificial neural networks. *CABIOS*, 8(1):39–44, 1992.

[Fetrow and Bryant, 1993] Jacquelyn S. Fetrow and Stephen H. Bryant. New programs for protein tertiary structure prediction. *Bio/Technology*, 11:479–484, 1993.

[Fienberg, 1977] Stephen E. Fienberg. *The Analysis of Cross-Classified Categorical Data.* MIT Press, 1977.

[Finer-Moore and Stroud, 1984] Janet Finer-Moore and Robert M. Stroud. Amphipathic analysis and possible formation of the ion channel in an acetylcholine receptor. *Proceedings of the National Academy of Sciences USA*, 81:155–159, 1984.

[Finkelstein and Nakamura, 1993] Finkelstein and Nakamura. Weak points of antiparallel beta-sheets: how are they filled up in globular proteins? *Advances in Gene Technology*, page 126, 1993.

[Flores *et al.*, 1993] T. P. Flores, C. A. Orengo, D. S. Moss, and J. M. Thornton. Comparison of conformational characteristics in structurally similar protein pairs. *Protein Science*, 2:1811–1826, 1993.

[Garnier *et al.*, 1978] Garnier, Osguthorpe, and Robson. Analysis of the accuracy and implications of simple methods for predicting the secondary structure of globular proteins. *Journal of Molecular Biology*, 120:97–120, 1978.

[Gibrat *et al.*, 1987] J.-F. Gibrat, J. Garnier, and B. Robson. Further developements of protein secondary structure prediction using information theory: New parameters and consideration of residue pairs. *Journal of Molecular Biology*, 198:425–443, 1987.

[Godzik and Skolnick, 1992] Adam Godzik and Jeffrey Skolnick. Sequence-structure matching in globular proteins: application to supersecondary and tertiary structure determination. *PNAS USA*, 89:12098–12102, 1992.

[Godzik *et al.*, 1992] Godzik, Skolnick, and Kolinksi. Simulations of the folding pathway of triose phosphate isomerase-type alpha/beta barrel proteins. *Proceedings of the National Academy of Sciences*, 89:2629–2633, 1992.

[Goodman, 1971] L. A. Goodman. The analysis of multidimensional contingency tables: stepwise procedures and direct estimation methods for building models for multiple classifications. *Technometrics*, 13:33–61, 1971.

[Hagler and Honig, 1978] Arnold T. Hagler and Barry Honig. On the formation of protein tertiary structure on a computer. *Proceedings of the National Academy of Sciences*, 75(2):554–558, 1978.

[Hayes-Roth and others, 1986] Barbara Hayes-Roth et al. Protean: Deriving protein structure from constraints. In *Proceedings of the AAAI Fifth National Conference on Artificial Intelligence*, pages 904–909, Los Altos, California, 1986. Morgan Kaufman.

[Hendlich *et al.*, 1990] Manfred Hendlich, Peter Lackner, Sabine Weitchus, Hannes Floeckner, Rosina Froschauer, Karl Gottsbacher, Georg Casari, and Manfred J. Sippl. Identification of native protein folds amongst a large number of incorrect models: The calculation of low energy conformations from potentials of mean force. *Journal of Molecular Biology*, 216:167–180, 1990.

[Hobohm *et al.*, 1992] Uwe Hobohm, Michael Scharf, Reinhard Schneider, and Chris Sander. Selection of reprsentative data sets. *Protein Science*, 1:409–417, 1992.

[Holley and Karplus, 1989] L. Howard Holley and Martin Karplus. Protein secondary structure prediction with a neural network. *Proc. Natl. Acad. Sci. USA*, 86:152–156, 1989.

[Holm and Sander, 1992] Liisa Holm and Chris Sander. Evaluation of protein models by atomic solvation preference. *Journal of Molecular Biology*, 225:93–105, 1992.

[Johnson *et al.*, 1993] Mark S. Johnson, John P. Overington, and Tom L. Blundell. Alignment and searching for common protein folds using a data bank of structural templates. *Journal of Molecular Biology*, 231:735–752, 1993.

[Jones *et al.*, 1992] D. T. Jones, W. R. Taylor, and J. M. Thornton. A new approach to protein fold recognition. *Nature*, 358:86–89, 1992.

[Kabsch and Sander, 1983] Wolfgang Kabsch and Christian Sander. Dictionary of protein secondary structure: Pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers*, 22:2577–2637, 1983.

[Kabsch and Sander, 1984] Kabsch and Sander. On the use of sequence homologies to predict protein structure: identical pentapeptides can have completely different conformations. *Proceedings of the National Academy of Sciences USA*, 81:1075–78, 1984.

[Karplus and Petsko, 1990] Martin Karplus and Gregory A. Petsko. Molecular dynamics simulations in biology. *Nature*, 347:631–639, 1990.

[Kelley and Holladay, 1987] L. Kelley and L. A. Holladay. Comparison of scales of amino acid side chain properties by conservation during evolution of four proteins. *Protein Engineering*, 1(2):137–140, 1987.

[Kendrew and others, 1958] J. C. Kendrew et al. A three-dimensional model of the myoblogin molecule obtained by x-ray analysis. *Nature*, 181:662–666, 1958.

[King, 1988] Ross D. King. *A machine learning approach to the problem of predicting a protein's secondary structure from its primary structure.* PhD thesis, University of Strathclyde, 1988.

[Kneller *et al.*, 1990] DG Kneller, FE Cohen, and R Langridge. Improvements in protein secondary structure prediction by an enhanced neural network. *Journal of Molecular Biology*, 214:171–182, 1990.

[Kraulis, 1991] Per Kraulis. Molscript: a program to produce both detailed and schematic plots of protein structures. *Journal of Applied Crystallography*, 1991.

[Kreusch *et al.*, 1994] A. Kreusch, A. Neubuser, E. Schiltz, J. Weckesser, and G. E. Schulz. Structure of the membrane channel porin from rhodopseudomonas blastica at 2.0 a resolution. *Protein Science*, 3:58–63, 1994.

[Kuntz *et al.*, 1976] I.D. Kuntz, G.M. Crippen, P.A. Kollman, and D. Kimelman. Calculation of protein tertiary structure. *Journal of Molecular Biology*, 160:983–994, 1976.

[Kyte and Doolittle, 1982] Jack Kyte and Russell F. Doolittle. A simple method for displaying the hydropathic character of a protein. *Journal of Molecular Biology*, 157:105–132, 1982.

[Lathrop and Smith, 1994] R. H. Lathrop and T. F. Smith. A branch-and-bound algorithm for optimal protein threading with pairwise (contact potential) amino acid interactions. In *Proceedings fo the 27th Hawaii International Conference on Systems Sciences*, pages 365–374, Los Alamitos, California, 1994. IEEE Computer Society Press.

[Lathrop *et al.*,] R. H. Lathrop, B. K. M. Bryant, L. Buturovic, L. Lo Conte, R. Nambudripad, S. Rao, J. V. Whtie, M. Cline, D. Haussler, A. Lapedes, and T. F. Smith. A comparison of gapped pairwise threading potentials. in preparation.

[Lathrop, 1994] R. H. Lathrop. The protein threading problem with sequence amino acid interaction preferences is np-complete. *Protein Engineering*, 7(9):1059–1068, 1994. in press.

[Lee and Subbiah, 1991] Christopher Lee and S. Subbiah. Prediction of protein side-chain conformation by packing optimization. *Journal of Molecular Biology*, 217:373–388, 1991.

[Leighton, 1993] Russell R. Leighton. *The Aspirin/MIGRAINES Neural Network Software User's Manual.* MITRE Corporation, 13706 Penwith Court, Chantilly, Virginia 22021, release v6.0 edition, October 1993.

[Lenstra, 1977] J. A. Lenstra. Evaluation of secondary structure prediction in proteins. *Biochimica et Biophysica Acta*, 491:333–338, 1977.

[Levin *et al.*, 1993] Jonathan M. Levin, Stefano Pascarella, Patrick Argos, and Jean Garnier. Quantification of secondary structure prediction improvement using multiple alignments. *Protein Engineering*, 6(8):849–854, 1993.

[Levinthal, 1968] C. Levinthal. Are there pathways for protein folding? *Journal of Chemical Physics*, 65:44–45, 1968.

[Levitt and Warshel, 1975] Michael Levitt and Arieh Warshel. Computer simulation of protein folding. *Nature*, 253:694–698, 1975.

[Levitt, 1978] M. Levitt. Conformational preferences of amino acids in globular proteins. *Biochemistry*, 17:4277–4285, 1978.

[Lifson and Roig, 1961] S. Lifson and A. Roig. On the theory of the helix-coil transition in polypeptides. *Journal of Chemical Physics*, 34:1963, 1961.

[Lifson and Sander, 1979] Shneior Lifson and Christian Sander. Antiparallel and parallel beta-strands differ in amino acid residue preferences. *Nature*, 282:109–111, 1979.

[Lifson and Sander, 1980] Shneior Lifson and Christian Sander. Specific recognition in the tertiary structure of beta-sheets of proteins. *Journal of Molecular Biology*, 139:627–639, 1980.

[Lim, 1974] V. I. Lim. Algorithms for prediction of alpha-helical and beta-structural regions in globular proteins. *Journal of Molecular Biology*, 88:837–894, 1974.

[Luthy *et al.*, 1991] Roland Luthy, Andrew D. McLachlan, and David Eisenberg. Secondary structure-based profiles: Use of structure-conserving scoring tables in searching protein sequence databases for structural similarities. *Proteins*, 10:229–239, 1991.

[Luthy *et al.*, 1992] Roland Luthy, Bowie, and David Eisenberg. Assessment of protein models with three-dimensional profiles. *Nature*, 356:83–85, 1992.

[Luthy *et al.*, 1994] Roland Luthy, Ioannis Xenarios, and Philipp Bucher. Improving the sensitivity of the sequence profile method. *Protein Science*, 3:139–146, 1994.

[Maclin and Shavlik, 1991] Richard Maclin and Jude W. Shavlik. Refining algorithms with knowledge-based neural networks. Machine Learning Research Group Working Paper 91-2, Univ. Wisc., Computer Science Department, 1991.

[Matouschek *et al.*, 1990] Andreas Matouschek, James T. Kellis, Luis Serrano, Mark Bycroft, and Alan R. Fersht. Transient folding intermediates characterized by protein engineering. *Nature*, 346:440–445, 1990.

[McGregor *et al.*, 1989] Malcolm J McGregor, Tomas P. Flores, and Michael JE Sternberg. Prediction of beta-turns in proteins using neural networks. *Protein Engineering*, 2(7):521–526, 1989.

[McLachlan and Karn, 1983] Andrew D. McLachlan and Jonathan Karn. Periodic features in the amino acid sequence of menatode myosin rod. *Journal of Molecular Biology*, 164:605–626, 1983.

[Metfessel and others, 1993] Metfessel et al. Cross-validation of protein structural class prediction using statistical clustering and neural networks. *Protein Science*, 2:1171–1182, 1993.

[Miyazawa and Jernigan, 1985] Sanzo Miyazawa and Robert L. Jernigan. Estimation of effective interresidue contact energies from protein crystal structures: Quasi-chemical approximation. *Macromolecules*, 18:534–552, 1985.

[Myers, 1991] Eugene Myers. An overview of sequence comparison algorithms in molecular biology. Technical Report TR 91-29, University of Arizona, Deparatment of Computer Science, Univ. of Ariz., Tucson, AZ 85721, 1991.

[Nall, 1986] Barry T. Nall. Native or nativelike species are transient intermediates in folding of alkaline iso-2 cytochrome c. *Biochemistry*, 25:2974–2978, 1986.

[Nambudripad *et al.*,] R. Nambudripad, L. Buturović, L. Tucker-Kellogg, S. Rao, and T. Smith. Development and characterization of protein core library. in preparation.

[Narayana and Argos, 1984] Narayana and Argos. residue contacts in protein structures and implications for protein folding. *IJPPR*, 24:25–39, 1984.

[Needleman and Wunsch, 1970] S. B. Needleman and C. D. Wunsch. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology*, 48:443–453, 1970.

[Niermann and Kirschner, 1990] Thomas Niermann and Kasper Kirschner. Improving the prediction of secondary structure of tim-barrel enzymes. *Protein Engineering*, 4(3):359–370, 1990.

[Nishikawa, 1983] K. Nishikawa. Assessment of secondary-structure prediction of proteins: comparison of computerized chou-fasman method with others. *Biochimica et Biophysica Acta*, 748:285–299, 1983.

[Ouzounis *et al.*, 1993] Christos Ouzounis, Chris Sander, Michael Scharf, and Reinhard Schneider. Prediction of protein structure by evaluation of sequence-structure fitness. *Journal of Molecular Biology*, 232:805–825, 1993.

[Overington *et al.*, 1992] Overington, Donnelly, Johnson, Sali, and Blundell. Environment-specific amino acid substitution tables: tertiary templates and prediction of protein folds. *Protein Science*, 1:216–226, 1992.

[Pascarella and Argos, 1991] Stefano Pascarella and Patrick Argos. Conservation of amphipathic conformations in multiple protein structural alignments. *Protein Engineering*, 10:229–239, 1991.

[Pauling and Corey, 1951] Linus Pauling and Robert Corey. Configurations of polypeptide chains with favored orientations around single bonds: two new pleated sheets. *PNAS USA*, 37:729–740, 1951.

[Ponnuswamy *et al.*, 1980] P. K. Ponnuswamy, M. Prabhakaran, and P. Manavalan. Hydrophobic packing and spatial arrangement of amino acid residues in globular proteins. *Biochimica et Biophysica Acta*, 623:301–316, 1980.

[Presnell and Cohen, 1993] S. R. Presnell and F. E. Cohen. Artificial neural networks for pattern recognition in biochemical sequences. *Annual Review of Biophysics and Biomolecular Structure*, 22:283–298, 1993.

[Qian and Sejnowski, 1988] Ning Qian and Terrence J. Sejnowski. Predicting the secondary structure of globular proteins using neural network models. *Journal of Molecular Biology*, 202(856-884), 1988.

[Rooman and Wodak, 1988] Marianne J. Rooman and Shoshana J. Wodak. Identification of predictive sequence motifs limited by protein structure data base size. *Nature*, 335:45, 1988.

[Rost and Sander, 1993a] Burhkard Rost and Chris Sander. Prediction of protein secondary structure at better than 70% accuracy. *Journal of Molecular Biology*, 232:584–599, 1993.

[Rost and Sander, 1993b] Burkhard Rost and Chris Sander. Secondary structure prediction of all-helical proteins in two states. *Protein Engineering*, 6(8):831–836, 1993.

[Rumelhart *et al.*, 1986] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning internal representations by back-propagating errors. *Nature*, 323:533–536, 1986.

[Sander and Schneider, 1991] Chris Sander and Reinhard Schneider. Database of homology-derived protein structures and the structural meaning of sequence alignment. *Proteins*, 9:45–68, 1991.

[Scheraga, 1978] H. A. Scheraga. Use of random copolymers to determine the helix-coil stability constants of the naturally occurring amino acids. *Pure and Applied Chemistry*, 50:315, 1978.

[Schiffer and Edmundson, 1967] M. Schiffer and A. E. Edmundson. Use of helical wheels to represent the structures of proteins and to identify segments with helical potential. *Biophysical Journal*, 7:121–135, 1967.

[Sippl and Weitckus, 1992] Sippl and Weitckus. Detection of native-like models for amino acid sequences of unknown three-dimensional structure in a data base of known protein conformations. *Proteins*, 13:258–271, 1992.

[Sippl, 1990] Manfred J. Sippl. Calculation of conformational ensembles from potentials of mean force; an approach to the knowledge-based prediction of local structures in global structures. *Journal of Molecular Biology*, 213:859–883, 1990.

[Skolnick and Kolinski, 1990] Jeffrey Skolnick and Andrzej Kolinski. Simulations of the folding of a globular protein. *Science*, 250:1121–1125, 1990.

[Sternberg and Islam, 1990] Michael J. E. Sternberg and Suhail A. Islam. Local protein sequence similarity does not imply a structural relationship. *Protein Engineering*, 4(2):125–131, 1990.

[Stolorz *et al.*, 1991] Paul Stolorz, Alan Lapedes, and Yuan Xia. Predicting protein secondary structure using neural net and statistical methods. Technical Report LA-UR-91-15. Los Alamos National Lab, 1991.

[Stryer, 1988] Lubert Stryer. *Biochemistry, Third Edition.* W. H. Freeman, New York, 1988.

[Taylor, 1989] William R. Taylor. A template based method of pattern matching in protein sequences. *Prog. Biophys. Molec. Biol.*, 54:159–252, 1989.

[Vila *et al.*, 1991] J. Vila, R. L. Williams, M. Vasquez, and H. A. Scheraga. Empirical solvation models can be used to differentiate native from near-native conformations of bovine pancreatic trypsin inhibitor. *Proteins*, 10:199–218, 1991.

[von Heijne and Blomberg, 1978] von Heijne and Blomberg. Some global beta-sheet characteristics. *Biopolymers*, 17:2033–2037, 1978.

[Warme and Morgan, 1978a] Warme and Morgan. A survey of atomic interactions in 21 proteins. *Journal of Molecular Biology*, 118:289–304, 1978.

[Warme and Morgan, 1978b] Warme and Morgan. A survey of atomic interactions in 21 proteins. *Journal of Molecular Biology*, 118:273–287, 1978.

[Weissman and Kim, 1991] Jonathan S. Weissman and Peter S. Kim. Reexamination of the folding of bpti: predominance of native intermediates. *Science*, 253:1386–1393, 1991.

[Wertz and Scheraga, 1978] D. H. Wertz and H. A. Scheraga. Influence of water on protein strutcure. an analysis of teh preferences of amino acid residues for the inside or outside and for specific conformations in a protein molecule. *Macromolecules*, 11:9–15, 1978.

[White *et al.*, 1994] J. White, I. Muchnik, and T. F. Smith. Models for protein cores based on markov random fields. *Math. Biosciences*, November 1994. in press.

[Wickens, 1989] Thomas D. Wickens. *Multiway Contingency Tables Analysis for the Social Sciences.* Lawrence Erlbaum Associates, Inc., 365 Broadway, Hillsdale, NJ 07642, 1989.

[Wilmot and Thornton, 1988] C. M. Wilmot and J. M. Thornton. Analysis and prediction of the different types of beta-turn in proteins. *Journal of Molecular Biology*, 203:221–232, 1988.

[Wodak and Rooman, 1993] Shoshana J. Wodak and Marianne J. Rooman. Generating and testing protein folds. *Current Opinion in Structural Biology*, 3:247–259, 1993.

[Zimm and Bragg, 1959] B. H. Zimm and J. K. Bragg. Theory of the phase transition between helix and random coil in polypeptide chains. *Journal of Chemical Physics*, 31:526, 1959.