# Discovering Protein Sequence-Structure Motifs
# and
# Two Applications to Structural Prediction

by

Thomas Cheuk Kai Tang

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Mathematics
in
Computer Science

Waterloo, Ontario, Canada, 2004

I hereby declare that I am the sole author of this thesis.  This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

# Abstract

This thesis investigates the correlations between short protein peptide sequences and local tertiary structures. In particular, it introduces a novel algorithm for partitioning short protein segments into clusters of local sequence-structure motifs, and demonstrates that these motif clusters contain useful structural information via two applications to structural prediction.

The first application utilizes motif clusters to predict local protein tertiary structures. A novel dynamic programming algorithm that performs comparably with some of the best existing algorithms is described.

The second application exploits the capability of motif clusters in recognizing regular secondary structures to improve the performance of secondary structure prediction based on Support Vector Machines. Empirical results show significant improvement in overall prediction accuracy with no performance degradation in any specific aspect being measured.

The encouraging results obtained illustrate the great potential of using local sequence-structure motifs to tackle protein structure predictions and possibly other important problems in computational biology.

# Acknowledgements

I would like to thank my supervisor Dr. Ming Li for accepting me into the program and guiding me along the way with patience, generosity, and sound advice. I would also like to thank my co-supervisor Dr. Jinbo Xu for offering bright suggestions and ample assistance that have helped me carry through the research. In addition, I would like to thank Dr. Forbes Burkowski for teaching me the basic techniques in structural bioinformatics and reviewing my thesis, Dr. Ian Munro for strengthening my background in data structures and also for reviewing my thesis, and finally Dr. Dan Brown for teaching me the fundamentals in sequence analysis.

My fellow graduate students have also helped me out a lot: Jun Chen has been very friendly and supportive throughout; Sriram Darbha has been a wonderful partner for all kinds of discussion; and Tomas Vinar has been a very professional technical support.

Finally, I have to thank my family and friends for being on my side. A special thank-you goes to my best buddy John Tsang for introducing me into the field and giving me practical career advice. My most sincere gratitude goes to Elaine Chan and my mom, for this thesis would not have existed at all without their unlimited understanding, patience, and support.

This research was partially funded by the PGS-A Scholarship granted by NSERC, and the Graduate Incentive Award granted by the University of Waterloo.

# Table of Contents

# List of Tables

# List of Illustrations

# Chapter 1

# Introduction

## 1.1 Importance of Protein Tertiary Structures

Every cell of every eukaryotic organism contains a copy of the blueprint for that organism: pairs of long and massive polymers stored in the form of a twisted double helix called *DNA* (deoxyribonucleic acid). Certain regions along DNA are called *genes*. Special signals found inside and in the vicinity of genes flag the cell to transcribe the genes into *RNA* (ribonucleic acid). While some genes code for RNA that is used directly by the cell for vital enzymatic purposes, most genes are *protein-coding*. That is, the resultant RNA is to be translated into another kind of polymer called *proteins*, which are ultimately responsible for the large majority of life functions. Despite their functional variety, all proteins are essentially created by chaining molecules called *amino acids* in different orders. The sequence of amino acids forms the *primary structure* of a protein. Each amino acid in a protein is called an *amino acid residue* or just *residue*, as its flanking atoms have been stripped off during the translation process.

The primary structure does not give the protein the ability to perform its functions right away. Instead, it causes the protein to undergo a *folding process*, in which the protein folds into a particular three dimensional (3D) shape believed to be the most energetically stable after taking into consideration all interactions among its residues. This shape, called the *tertiary structure*, enables the protein to interact with other proteins and/or molecules in order to achieve its intended biological functions.

Biologists have long realized the utmost importance of the tertiary structures of proteins. Simply put, the tertiary structure dictates how well a protein carries out its activities. Improperly folded proteins may lose their functions entirely or even assume new but undesirable ones, as in the case of Bovine Spongiform Encephalopathy, commonly known as Mad Cow disease. Other common lethal diseases resulting from protein misfolding include Alzheimer’s disease, Parkinson’s disease, and type II diabetes, among others. Therefore, knowing the shape of a protein is not only vital in understanding its biological roles, but also in developing possible cures should the protein misfold or disappear for any reason.

## 1.2 Challenges in Protein Tertiary Structure Prediction

It has been shown more than 30 years ago that all the information needed for a protein to fold resides in its amino acid sequence[1] [1]. Unfortunately, while current technologies such as gene-finding and mRNA micro-arrays have given us ample access to novel protein sequences, finding tertiary structures given the sequences remains a daunting challenge. Laboratory methods such as NMR Microscopy and X-ray Crystallography do exist for fold determination, but they are expensive and time-consuming. Even worse, the methods fail for proteins that are difficult to

[1] Exceptions to the rule such as folds created with the aid of chaperons or post-translational modifications are generally ignored for simplification purposes.

crystallize, especially membrane proteins. On the computational side, accurate prediction of tertiary structure is still out of reach after decades of research. Owing to its urgency and substantial impact on mankind, protein tertiary structure prediction is considered one of the most critical problems in computational biology.

A major obstacle for protein tertiary structure prediction lies in the complexity of modeling protein 3D conformations due to the large degree of structural freedom and sophisticated interactions among residues. Previous computational approaches include a number of lattice and off-lattice models as surveyed by Yuan et al. [2], all of which essentially formulate structural prediction into a large-scale search problem with limited success. Inspired by the conjecture that a newly created polypeptide forms local folds in parts before settling to its final fold [3], a model has recently emerged that treats a protein as a composition of local structural motifs. This model manages to reduce the size of protein conformational space to a point where many search-based prediction strategies finally become feasible. As a result, extraction of local motifs has always been a subject of intense study (see Section 2.1 for examples).

The tertiary structure of a protein is a concerto of two kinds of residue interactions: long-range interactions between distant residues such as disulfide bridges and inter-group charges, and local interactions among nearby residues. Xu et al. [4] have created RAPTOR, an innovative protein tertiary structure predictor based on optimal threading by linear programming. Unfortunately, as RAPTOR focuses primarily on achieving optimal global mapping between target and homologous proteins, it lacks a mechanism for refining output predictions based on local sequence patterns. This shortcoming has led to the investigation of local protein folds and their potential in *ab initio local structure prediction*, the prediction of tertiary structures of short protein segments based solely on the sequence information contained in the segments (See Section 2.2 for further details).

## 1.3 Some Biology Background

### 1.3.1 Protein Structure

A protein is a polymer consisting of many repeatedly linked units called amino acids. All amino acids (except proline) have the structure shown in Figure 1.1. In general, each amino acid has a central alpha-carbon atom $C_\alpha$ connected to a hydrogen atom, an amino group ($NH_2$), a carboxyl group (COOH), and a *side chain* denoted by R in the diagram. Because the carboxyl group is characteristic of all organic acids, the simultaneous presence of the amino group and the carboxyl group gives rise to the name “amino acid”.

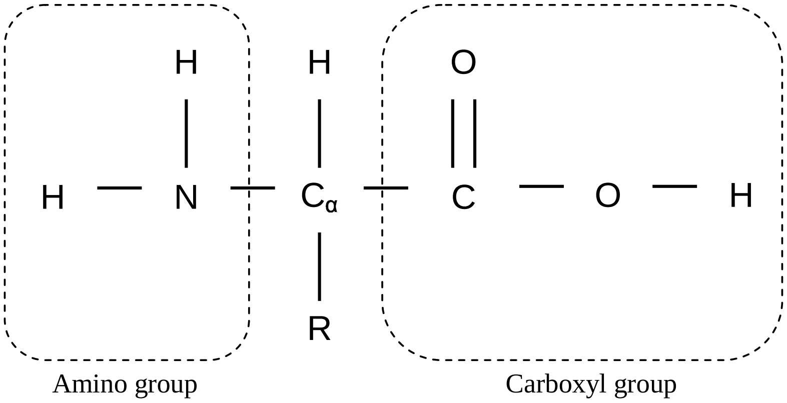


Figure 1.1: General structure of an amino acid (except praline) with side chain R

There are 20 standard amino acids distinguished by different side chain configurations. Other non-standard amino acids exist, but they are rare and only found in organisms inhabiting extreme environments such as volcanoes and ocean bottoms. Therefore, these non-standard amino acids are irrelevant as far as the majority of the research, including this thesis, is concerned. Figure 1.2 lists the names, symbols, and molecular structures for all 20 standard amino acids.

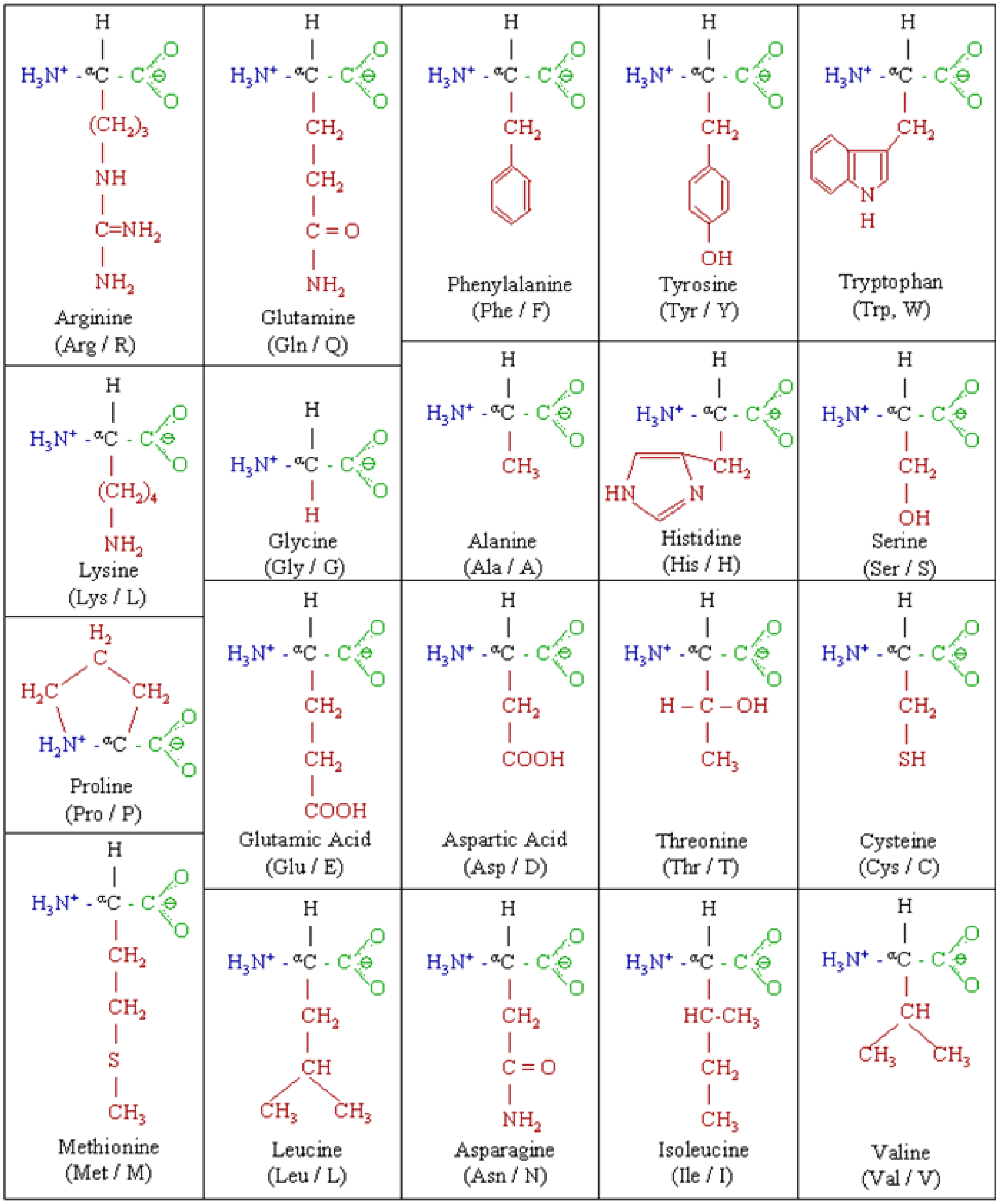


Figure 1.2: The 20 amino acids (taken from http://en.wikipedia.org/wiki/Amino_acid)

The difference in side chain configuration induces different properties on each amino acid. For instance, an amino acid can be *hydrophilic* (water-loving) or *hydrophobic* (water-repelling), polar or non-polar, charged or neutral, flexible or rigid, etc. Table 1.1 categorizes the amino acids based on different properties.

Table 1.1: Properties of standard amino acids

| **Properties** | **Property Description** | **Amino Acids** |
|---|---|---|
| Hydrophobic | Hydrophobic amino acids stay inside of a protein, while hydrophilic ones tend to stay in the exterior. | V, L, I, M, F |
| Hydrophilic | | N, E, Q, H, K, R, D |
| In-between | | G, A, S, T, Y, W, C, P |
| Positively charged | Oppositely charged amino acids can form salt bridges. | R, H, L |
| Negatively charged | | D, E |
| Polar but not charged | Polar amino acids can participate in hydrogen bonding. | N, Q, S, T |
| Non-polar | | A, G, I, L, M, P, V |

Guided by a sequence of codons (triplets of nucleotides) in a RNA molecule, a cell organelle called the ribosome creates a protein by linking amino acids together with *peptide bonds*, as shown in Figure 1.3. Therefore, a protein is also called a *polypeptide* because it consists of many amino acid residues linked together by peptide bonds.

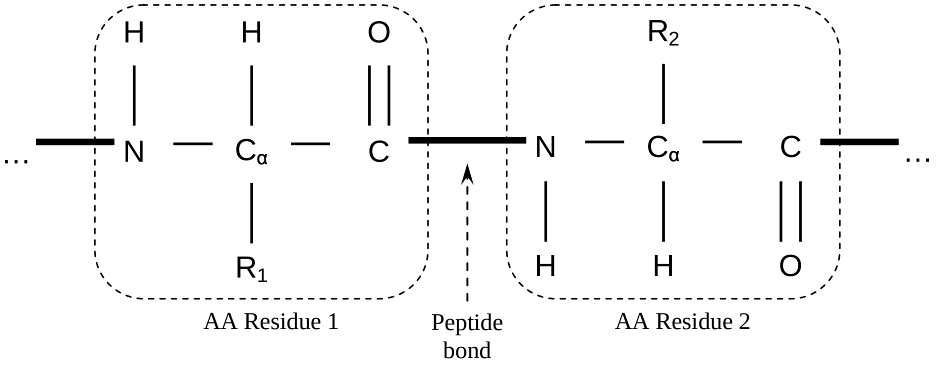


Figure 1.3: Peptide bond linking two amino acid (AA) residues

Although the entire 3D conformation of a protein can be expressed with coordinates for all atoms, it suffices to just consider the coordinates of the *backbone* atoms, namely the repeating (N, $C_\alpha$, C) chain of atoms in Figure 1.3. Scientists also use *dihedral angles* to precisely describe the shape of a protein. A dihedral angle is defined as the torsion angle from one planar surface to another, as depicted in Figure 1.4. Dihedral angles must be between $180^o$ and $-180^o$ inclusive.

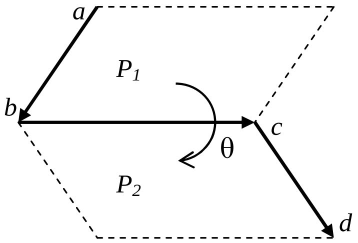


Figure 1.4: Dihedral angle θ from $P_1$ (defined by *ab* and *bc*) to $P_2$ (defined by *bc* and *cd*)

There are three types of dihedral angles associated with the backbone of a protein, namely the *phi* (φ), *psi* (ψ), and *omega* (ω) angle. Figure 1.5 depicts the different dihedral angles along a protein backbone. Note that the diagram also shows the three types of bonds connecting the backbone atoms: the C-N bond (i.e. the peptide bond), the N-$C_\alpha$ bond, and the $C_\alpha$-C bond. If ($b_1$, $b_2$) represents the plane defined by non-collinear bonds $b_1$ and $b_2$, then φ is the dihedral angle from (C-N, N-$C_\alpha$) to (N-$C_\alpha$, $C_\alpha$-C), ψ is the dihedral angle from (N-$C_\alpha$, $C_\alpha$-C) to ($C_\alpha$-C, C-N), and ω is the dihedral angle from ($C_\alpha$-C, C-N) to (C-N, N-$C_\alpha$). Since bond lengths and bond angles are fairly rigid under normal biological conditions [5], the series of backbone dihedral angles are sufficient to describe the full conformation of a protein.

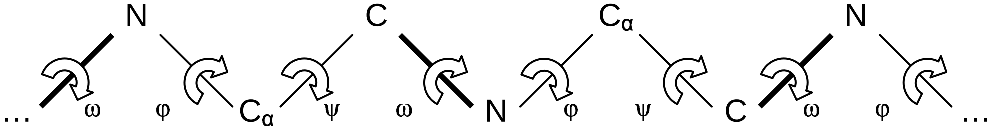


Figure 1.5: Dihedral angles along protein backbone (thick lines denote peptide bonds)

Further simplification is possible as peptide bonds can only exist in either the *cis* ($\omega = 0^o$) or the *trans* ($\omega = \pm180^o$) configuration. Peptide bonds in the *cis* configuration are a lot rarer, and their presence usually indicates special regional activities or structures due to their less energetically favorable nature [6]. Hence, $\omega$ angles are often assumed to be $180^o$ under normal circumstance, leaving only $\varphi$ and $\psi$ angles to express the whole protein geometry.

### 1.3.2 Sequence Profiles

Despite numerous possible mutations and re-arrangement events, certain genes are well conserved across species after a long period of time due to their important biological functions. Nevertheless, however conserved the genes are, their resultant proteins could have very different primary structures. Sander and Schneider [7], for example, have determined empirically that structure homology is implied even for proteins with as low as 25% sequence similarity[2]. Their study and others alike have confirmed the inadequacy of solely comparing primary structures for determining if two proteins are evolutionarily related. The correct alternative would be to compare sequence profiles instead. A *sequence profile* or *frequency profile* of a protein shows the probability of observing each amino acid in each position along the protein. It is generated from a multiple sequence alignment in which the protein is aligned to its homologues. The whole idea is that if two proteins are indeed evolutionarily related, then they must share the same ancestor and homologous siblings, and therefore similar sequence profiles.

There are many methods for generating sequence profiles, two of which are especially common in the research community. The first method is to use a tool called PSI-BLAST [8], a brief description of which can be found in Section 2.3.1. The second method is to generate profiles based on alignments available in the HSSP database [7]. Note that this method could

[2] Measured in an alignment over a length of 80 residues or longer

result in biased profiles as the alignments used might have been heavily populated by proteins from certain families but lightly so by others. Fortunately, many sequence weighting methods exist for correcting such unequal representation including simple pseudo-counts, Voronoi weights [9], Maximum discrimination weights [10], and Maximum entropy weights [11].

Rost and Sander [12] achieved a 6% increase in prediction accuracy in their neural network method when they replaced primary structures with sequence profiles for prediction. Improvement resulting from the usage of sequence profiles was also confirmed by de Breven et al. [13]. As implied in Jones' work [14], the quality of sequence profiles has a dramatic impact on performance.

## 1.4 Protein Motifs

Within the context of this thesis, a *motif* is defined as a recurrent feature shared by a significant number of segments that are extracted from proteins belonging to different families. There are three main categories of motifs, namely *sequence motifs*, *structural motifs*, and *sequence-structure motifs*.

As its name suggests, a sequence motif describes a recurrent sequence pattern found in a significant number of protein segments. Likewise, a structural motif describes a recurrent structural pattern. One might often be misled by intuition that sequence similarity automatically implies structural similarity, which would have been true if the folding of protein segments were solely determined by local inter-residue interactions within the segments. Unfortunately, there are also long-range interactions such as disulfide bridges, inter-group charges, and hydrophobic effects that alter the overall tertiary structure of a protein. Segments under the influence of such global forces would fold differently from other segments even if they share a high degree of sequence similarity. Since it is ultimately the shape that matters to the protein's capability, sequence motifs tend to be less valuable and much less frequently studied.

On the contrary, structural motifs are more intensively studied because they constitute the conformational search space for many search-based structural prediction algorithms [15, 16, 17]. Nevertheless, structural motifs neglect specific sequence information that characterizes their formation, so their usage in *ab initio* structural prediction usually requires some sort of external guidance such as a global energy function.

The last category comprises sequence-structure motifs. Each sequence-structure motif is shared by segments that are highly similar in both sequence composition and tertiary structure. By definition of motifs, these segments must amount to a significant number and belong to proteins from different families, so the observed sequence-structure correlation is almost impossible to happen by chance. As a result, one may deduce with high confidence that for any sequence-structure motif, the structure is mostly or even entirely determined by the corresponding sequence pattern. This important concept forms the underlying principle that enables sequence-structure motifs to map short sequence patterns into relevant structures.

## 1.5 Research Overview

The two main problems being addressed by this thesis are the discovery of sequence-structure motifs given a set of non-redundant proteins, and the prediction of protein folds by exploiting the motifs' ability to map sequences to structures. The rest of this thesis is organized as follows: Chapter 2 will present a detailed survey of some of the previous related work, Chapter 3 will describe a novel clustering algorithm for extracting sequence-structure motifs, Chapter 4 will describe a novel dynamic programming algorithm for *ab initio* local structure prediction using motif clusters, Chapter 5 will describe a procedure for using motif clusters to enhance secondary structure prediction based on Support Vector Machines, and finally Chapter 6 will conclude the paper with some final comments and a list of future work.

# Chapter 2

# Related Work

The materials presented in this thesis concern three major areas: motif extraction via clustering, *ab initio* local structure prediction, and secondary structure prediction. This section will present some background and related research in each area.

## 2.1 Extraction of Protein Motifs via Clustering

Clustering is a popular statistical technique for analyzing large data sets through the grouping of similar items. It enables researchers to focus on the general patterns instead of on the individual items themselves. Since motifs are patterns shared by a significant number of segments, their extraction can be properly achieved through clustering. Previous methods for clustering short protein segments are generally divided into two categories: 1) those with clustering based on structure alone, and 2) those with clustering based on both sequence and structure. A brief survey of six recent methods is presented below, where the first three belong to the first category and the rest belong to the second.

### 2.1.1 Clustering using Reference Frame O*xyz*

Wojcik et al. [15] invented a novel orthogonal reference frame called O*xyz* for aligning two loop segments in order to calculate their structural deviation. For each loop segment *s*, the O*x* axis was the line joining the $C_\alpha$ atoms of the first and last residues in *s*, O*y* axis was defined such that the plane formed by O*x* and O*y* contained the centre of gravity of the backbone of *s*, and finally O*z* axis was the vector product of O*x* and O*y*. Loop segments of length 3 to 8 residues long were extracted and classified using hierarchical clustering based on RMSD (root mean squared distance) between backbone atoms positioned in O*xyz*. Each resultant cluster essentially represented a structural motif. The set of all clusters was subsequently used for loop modeling in the remainder of the study.

### 2.1.2 Clustering using K-means Stimulated Annealing

Kolodny et al. [16] clustered protein segments of length 4 to 7 residues long using a modified k-means algorithm called *k-means stimulated annealing*, which was identical to the original k-means algorithm [18, 19] except that two clusters were merged and another was split in a Monte Carlo fashion at the end of each iteration. RMSD after superposition was used within the algorithm to measure the distance between any two given segments. The authors claimed that their special k-means algorithm improved the handling of segment concentrations and reduced sensitivity to the initial choice of cluster centers. The quality of the resultant clusters was evaluated by examining how well the clusters could fit into the structures of certain test proteins.

### 2.1.3 Clustering using Hypercosine as Distance Measure

Although RMSD is a popular measure of inter-segment distances, its usage in large-scale clustering could be hindered by its expensive numerical computation. Hunter et al. [17] thus suggested a new structural distance measure based on *hypercosine*, which they claimed was a good approximation of RMSD while being much more efficient to compute. To measure the structural deviation between two segments, a vector was created from each segment by taking the backbone atomic coordinates along the segment after it was aligned in the 3D Cartesian space with its first $C_\alpha$ atom at the origin, its last $C_\alpha$ atom on the *z* axis, and its second $C_\alpha$ atom on the *x*-*z* plane. Let *u* and *v* be the resultant vectors for the two segments, $\langle u, v \rangle$ be their inner product, and $|u|$ and $|v|$ be the $l^2$-norm (i.e. the magnitude) of *u* and *v* respectively. The hypercosine between *u* and *v*, denoted by *HCos*(*u*, *v*), was computed as follows:

$$HCos(u,v) = \frac{\langle u, v \rangle}{|u||v|} \tag{2.1}$$

The output was a real value in the range [0, 1], with 0 indicating totally structural dissimilarity and 1 indicating structural identicalness. Hunter and his peers tested the efficiency of the new method by clustering 150,000 length-7 segments. The remainder of their study focused on how changes in hypercosine threshold affected the quality of the resultant cluster set.

### 2.1.4 Protein Blocks

French scientists de Brevern et al. [13] clustered protein segments of length 5 based on both sequence and structure in a two-stage process. During the first stage, they partitioned segments by structure into 16 clusters called *Protein Blocks* as follows. Each segment centered at $C\alpha_i$ was represented as a vector of eight dihedral angles ($\psi_{i-2}$, $\varphi_{i-1}$, $\psi_{i-1}$, $\varphi_i$, $\psi_i$, $\varphi_{i+1}$, $\psi_{i+1}$, $\varphi_{i+2}$), and inter-segment distance was measured by RMSDA (root mean squared distance on angular values). An unsupervised Kohonen network formed the basis of the clustering method. In short, the method was initialized with a fixed number of randomly drawn cluster centroids. It then assigned each segment to its closest cluster and updated the cluster's centroid accordingly. Once all segments had been exhausted, the assignment process restarted with the new centroids. This was repeated for a given number of times to obtain the final set of clusters or Protein Blocks.

Every Protein Block was further sub-divided into *sequence families* based on sequence composition in the second stage. Initially, all segments within each Protein Block were arbitrarily partitioned into a fixed number of groups, and the consensus sequence profile for each group was computed. Each segment was then assigned to the group whose profile yielded the highest conditional probability of observing the segment. After all segments had been assigned, new profiles were generated for the groups and the assignment process was repeated. The whole procedure stopped when changes to new profiles were minimal. The resultant sequence families from all 16 Protein Blocks ultimately formed the complete set of sequence-structure motifs.

### 2.1.5 I-sites Library

The *I-sites Library*, created by Bystroff and Baker [20], is a collection of 13 different sequence-structure motifs commonly found in proteins. Similar to Protein Blocks, the motifs were also extracted via a two-stage process. In the first stage, segments of length 3 to 15 were partitioned based on sequence similarity using a custom distance function and the k-means clustering algorithm [18, 19]. A refinement process in the next stage removed segments whose structures were different from the paradigm structures of their respective clusters. The remaining segments in each cluster were combined to form a signature sequence profile used to further search for other similar segments in the training database. The segments that were initially removed from each cluster formed a new cluster, which then underwent the same refinement process. Because each repetition isolated segments sharing the paradigm or "peak" structure from the rest, the refinement process was called *iterative peak removal*. The end result was a set of 82 clusters that could be roughly grouped into 13 different sequence-structure motifs.

### 2.1.6 LPBSP1

Yang and Wang [21] created a *local structure-based sequence profile* database called *LPBSP1*. Local structure-based sequence profiles are equivalent to sequence-structure motifs as they both represent the consensus of a group of segments sharing similar compositions and structures.

Prior to clustering, each segment in the training database was preprocessed such that every *phi* and *psi* angle pair was converted into a *backbone conformational state* as defined by Oliva et al. [22]. This preprocessing step essentially mapped the continuous backbone structures into discrete states needed for subsequent local structure prediction (see Section 2.2.3).

Each local structure-based sequence profile was created by first selecting a segment known as the seed. To refine the seed, all segments in the training database were examined, and those resembling the seed's structure were extracted. The resultant segments were then filtered using a custom scoring matrix to retain those that were also sequence-wise similar to the seed. The final set of segments composed a consensus sequence profile, which was used along with the seed's structure to "fish" out other unidentified homologues in the training database. At the end, all segments that had been found constituted the sequence-structure motif cluster for the seed segment. The entire process was repeated for all segments in the database, and all resultant clusters with less than 10 segments were discarded. Yang and Wang applied the clustering method to 213,338 length-9 segments to obtain a final set of 138,604 clusters.

## 2.2 *Ab Initio* Local Structure Prediction

The aim of *ab initio* local structure prediction is to predict the tertiary structures of short protein segments based solely on the sequence information contained in the segments. A main driving force behind such prediction is its potential in tackling the protein folding problem [23], the resolution of which lends itself to other critical problems such as *ab initio* prediction of global tertiary structures and identification of protein misfolding.

Protein folding is very difficult to simulate mainly because of the many different ways residues can interact with their distant counterparts. These long-range interactions play a vital role in guiding polypeptides to fold properly upon their creation. Given the efficiency of the folding process, however, it is impossible for a polypeptide to consider all distant interactions or even a majority of them. As a result, the folding process is believed to initiate with segments folding locally, forming structural intermediaries whose interactions lead to the final shape. Therefore, the study of local structures and their formations would be a steppingstone, if not a prerequisite, to understanding the folding process.

Macromolecular structure repositories such as the Brookhaven Protein Databank (PDB) have enabled researchers to discover a number of local structural motifs, such as the Schellman motif [24], the hydrophobic staple [25], the extended capping box [26], and various beta-hairpin structures [27, 28]. The values of local motifs to structural prediction have been noted in a number of studies including Bonneau et al. [29], Fidelis et al. [30], and Rooman et al. [31]. The following sub-sections describe *ab initio* local structure prediction using Protein Blocks (Section 2.1.4), the I-sites Library (Section 2.1.5), and LPBSP1 (Section 2.1.6).

### 2.2.1 Prediction using Protein Blocks

After discovering Protein Blocks and sequence families, de Brevern et al. [13] went on to apply them to local structure prediction. In their prediction method, Protein Block $b$ was assigned to a target segment $s$ if sequence family $f \subseteq b$ yielded the largest ratio $r$ = P($f$ | $s$) / P($f$) among all sequence families[3]. The ratio $r$ was calculated using only the sequence profiles of $f$ and $s$. To predict the local structure of a target protein $p$, the method simply assigned the optimal protein block (i.e. the one yielding the largest ratio $r$) to each overlapping segment in $p$. During the evaluation process, an assignment involving Protein Block $b$ and segment $s$ was considered correct if $b$ was also the Protein Block structurally closest to the true conformation of $s$. The overall prediction accuracy, evaluated as the percentage of correct assignments over the total, was 40.7%. The authors subsequently claimed better accuracies by considering multiple top-scoring Protein Blocks, instead of just the optimal one, for each segment. Unfortunately, those results were practically meaningless because the authors failed to instruct which Protein Block was to be chosen should the true structures be unavailable for comparison, as in a real prediction scenario.

### 2.2.2 Prediction using the I-sites Library

As described in Section 2.1.5, the I-sites library is a collection of 82 clusters grouped into 13 sequence-structure motifs. Recall that during the extraction process, Bystroff and Baker [20] utilized a custom distance function to cluster segments based on sequence similarity. The same distance function was used to score sequence similarity between a cluster and a given segment during local structure prediction. Since different clusters were of different lengths, the similarity scores were not directly comparable. As a remedy, Bystroff and Baker mapped each score into a

[3] P($x$) is the probability of observing $x$, and P($x$ | $y$) is the probability of observing $x$ given the presence of $y$.

confidence value, which stood for the likelihood of the segment having the structure of the cluster given the score. The mapping was derived empirically through a cross-validation procedure.

The prediction method first computed the confidence values of all overlapping segments in the target protein versus all 82 clusters. The set of all segment-cluster pairs were then sorted by confidence values in descending order. The first segment-cluster pair was processed by assigning the consensus dihedral angles of the cluster to the residues in the segment. Each subsequent segment-cluster pair was processed only if the consensus dihedral angles of the cluster did not conflict with the ones previously assigned to the segment.

A residue was correctly predicted if it was found in at least one length-8 segment whose predicted structure was within 1.4 Å in RMSD of the true structure. The overall prediction accuracy, evaluated as the percentage of correctly predicted residues over the total, was 50%.

### 2.2.3 Prediction using LPBSP1

LIBSP1 [21], a collection of 138,604 sequence-structure motif (see Section 2.1.6), was created specifically for local structure prediction. The sequence composition for each motif was represented by a position specific scoring matrix (PSSM) created with the Bayesian prediction pseudo-count method [32]. To predict the structure of a lengh-9 segment *s*, Yang and Wang searched through LIBSP1 to obtain the set *W* of all motifs whose PSSMs yielded high similarity scores for *s*, and assigned the structure of the motif located at the center of *W* to *s*. The above process was repeated for each overlapping length-9 segment along the target protein. At the end, the final prediction for each residue was taken to be the majority conformation found in the 9 overlapping predictions covering the residue. Yang and Wang developed an evaluation scheme called *RMSDaccuracy*, which they claimed was comparable to the RMSD measure used by Bystroff and Baker [20]. Their published result under the scheme was 62.1%.

Unfortunately, the method did not achieve high accuracy without cost. Because of the need to find the majority conformation, the continuous backbone dihedral angles were mapped into discrete states, as described in Section 2.1.6. The final predicted structure was expressed as a string of only four states {A, B, G, E}, so it was at best a rough approximation. In other words, there had been a trade-off between prediction accuracy and preciseness of predicted structures.

## 2.3 Enhancement to Secondary Structure Prediction

The tertiary structure of a protein can be seen as a spatial arrangement of three types of 3D sub-structures known as helices, strands, and coils. The distribution of these sub-structures along a protein is referred to as the *secondary structure* of the protein. While *ab initio* prediction of tertiary structure is difficult, that of secondary structure is a lot simpler because the latter projects the complicated 3D structures onto a linear sequence of H (helix), E (strand), and C (coil). Knowledge of secondary structures is often used as a constraint to tertiary structure prediction or as part of fold recognition methods [33]. There are numerous *ab initio* secondary structure prediction methods such as BRNN [34], DSC [35], NNSSP [36], PHD [37], PREDATOR [38], SVM [39], and Zpred [40]. Given the array of methods, a more practical option would be to enhance the performance of the best in the herd. Two example attempts to be described in this section are PSIPRED [14] and PMSVM [41].

### 2.3.1 PSIPRED

PSIPRED [14] is considered an improved version of PHD [37], a predictor widely recognized for its supreme accuracy. The main improvement comes from the use of position specific scoring matrices (PSSMs) generated by PSI-BLAST [8]. Given a query sequence, PSI-BLAST searches for high-scoring homologues from a non-redundant protein database, creates a profile from the homologues, and repeats the search with the new profile. The process lasts for a specified number of iterations. The utilization of PSI-BLAST profiles has increased the accuracy (or $Q_3$ to be exact, see Section 5.4) by about 5% on average, from around 73% to around 78%. At present, PSIPRED remains one of the most reliable secondary structure prediction methods available.

### 2.3.2 PMSVM

*Support Vector Machine* [42], or *SVM* for short, is a powerful statistical method for data classification. The most common use of SVM is as a binary classifier. In a nutshell, training a binary SVM classifier involves computing the separating hyper-plane that divides the training data points in such a way as to achieve maximal margin (i.e. to maximize the gap between the plane and the closest data points on either side). Once trained, new data points are classified to either category depending on which side of the hyper-plane they land on.

Hua and Sun [39] invented a secondary structure prediction method based on SVM, and achieved prediction accuracies that rivaled PHD, if not better. Motivated by the success of PSIPRED, Guo et al. [41] set out to improve the SVM prediction method of Hua and Sun. Besides utilizing PSI-BLAST profiles, they introduced a second SVM prediction layer to produce a dual-layer SVM predictor called PMSVM. The second layer was meant to refine the output of the first by considering the patterns of surrounding secondary structures for each residue. Guo et al. reported around 79% as the average prediction accuracy ($Q_3$) for PMSVM, an improvement of about 5% over the single-layer SVM approach.

# Chapter 3

# Discovery of Sequence-Structure Motifs

Clustering of short protein segments will be used as the primary approach for the discovery, or extraction, of sequence-structure motifs. Many of the previous methods, such as those described in Section 2.1.4 and Section 2.1.5, perform clustering in two stages. A problem associated with a two-stage approach is that segments with similar sequence patterns and folds might not as clearly reveal such a relationship when one looks at sequence and structure separately. Those segments are likely to get misclassified in either or both stages. This section presents a novel one-stage method intended to eliminate the deficiency by considering both sequence and structure together throughout the whole clustering process. Specifically, this section describes the inter-segment distance measure, segment preparation and filtration, the main clustering algorithm, and the experiments conducted and results gathered.

## 3.1 Segment Attributes

All protein segments are assumed to be of the same length *L*. Every segment is represented as an array of *L* records, each of which stores information for one residue. The stored information includes the occurrence frequencies for all 20 amino acids, the secondary structure label (H, E, or C), and all three backbone dihedral angles in degrees. Backbone atomic coordinates are not explicitly stored, but are calculated as needed from the dihedral angles. Table 3.1 below lists the values of bond lengths and bond angles determined empirically by Engh and Huber [43].

Table 3.1: Bond lengths and bond angles along protein backbone

| | |
|---|---|
| N-$C_\alpha$ bond length | 1.458 Å |
| $C_\alpha$-C bond length | 1.525 Å |
| C-N bond length | 1.329 Å |
| N-$C_\alpha$-C bond angle | 111.2$^o$ |
| $C_\alpha$-C-N bond angle | 116.2$^o$ |
| C-N-$C_\alpha$ bond angle | 121.7$^o$ |

## 3.2 Measure of Inter-Segment Distance

Each of the 20 amino acids is represented by a unique index in the range 0 to 19 inclusive. The exact index assignment is irrelevant but it must be consistent throughout the study. Let $\varphi_i$ and $\psi_i$ be the *phi* and *psi* angles in degrees at position *i*, and $f_{ij}$ be the frequency of observing amino acid with index *j* at the same position. Note that the condition $\sum_{j=0,19} f_{ij} = 1$ must hold for all *i*. Given segments *x* and *y*, both of length *L*, their distance *D*(*x*, *y*) is computed as follows:

$$D(x,y)=\begin{cases}\sqrt{\sum_{i=0}^{L-1}\left(\left(\frac{\Delta\varphi_i}{360}\right)^2+\left(\frac{\Delta\psi_i}{360}\right)^2+\sum_{j=0}^{19}\Delta f_{ij}{}^2\right)} & \text{if } \max(\Delta\varphi_i,\Delta\psi_i)\le\theta\ \ \forall\, i\\ \infty\ \text{ otherwise}\end{cases}\tag{3.1}$$

Symbol Δ denotes the absolute difference in the associated quantity. Value $\theta$ is *L*-dependent and it limits the largest dihedral angle difference allowed. Note that Equation (3.1) has two ideal properties as a distance function. First, it encompasses differences in both sequence patterns and structures, hence allowing one-stage clustering. Second, it is the Euclidean distance between two points in a 22*L*-dimensional Cartesian space and therefore satisfies the triangular inequality, making it acceptable for use in clustering [44].

The validity of Equation (3.2) below justifies the assumption that contributions from differences in structure and in sequence have equal weights. The second condition as well as the tightness of both bounds can all be proven trivially.

$$0 \leq \left(\frac{\Delta\varphi_i}{360}\right)^2 + \left(\frac{\Delta\psi_i}{360}\right)^2 \leq 2 \text{ and } 0 \leq \sum_{j=0}^{19} \Delta f_{ij}{}^2 \leq 2 \quad \forall\, i \in [0, L-1] \tag{3.2}$$

## 3.3 Cluster Radius

Besides a distance function, a threshold called *cluster radius* is needed to tell if two segments are sufficiently close to be grouped together. The choice of cluster radius is crucial: being too small yields a handful of clusters capturing only the most conserved motifs, while being too big yields coarse clusters contaminated with irrelevant segments. A systematic way exists to determine a suitable radius for a given segment length. First, segments of that length are extracted from a large database of non-redundant proteins whose structures are known. An ideal choice for the database would be PDB Select 25 [45, 46]. The set of all segments are then divided in half, and distances between segments in different halves are computed. The resultant figures form a normal distribution with mean $\mu$ and standard deviation $\sigma$, as shown in Figure 3.1 for segments of length 8. The radius is set to $\mu - 3\sigma$, corresponding to a confidence interval of 99.73%. This choice of radius is found to consistently deliver clusters of reasonable quality.

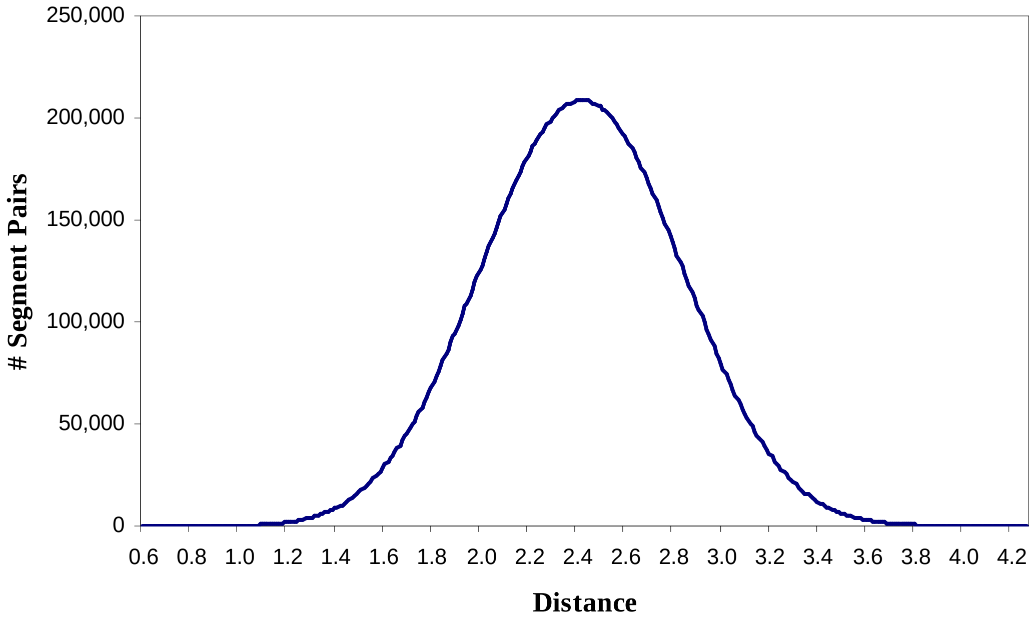


Figure 3.1: Normal distribution for inter-segment distances obtained from a large sample of length-8 segments, with mean $\mu$ = 2.42 and standard deviation $\sigma$ = 0.41

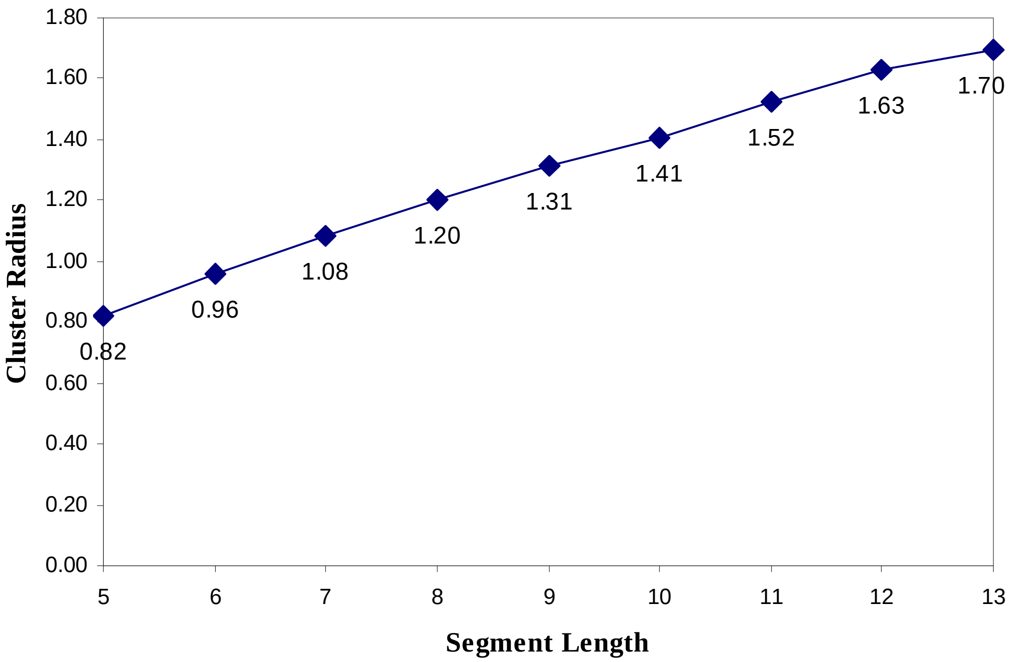


Figure 3.2: Fluctuation in cluster radius as segment length increases from 5 to 13

The cluster radii are larger for longer segments because they have to account for differences between more residues. Intuitively, once the segment length doubles, so does the radius for having to account for differences between twice as many residues. Therefore, a roughly linear relationship is expected between the radius and the segment length. Our empirical method for radius determination seems legitimate in that it does produce results that agree with expectation. Figure 3.2 shows the increase in cluster radius as segment length increases from 5 to 13 inclusive.

## 3.4 Segment Preparation and Filtration

The distance function shown in Equation (3.1) requires sequence profiles for both segments stating the occurrence frequency of each amino acid at every position. The profiles in this study are generated from multiple sequence alignments available in the HSSP database [7], and post-processed with the Voronoi Monte Carlo algorithm [9] to correct for unequal representations. Aside from profiles, secondary structure labels are also gathered, and for that the DSSP secondary structure labeling [47] is chosen due to its popularity.

A filtration process is in place to ensure the legitimacy of segments used for clustering. Specifically, a segment is not qualified unless it meets all three requirements: it must be aligned to at least 20 proteins in the HSSP database, comprise only standard residues, and contain only *trans* peptide bonds between residues (see Section 1.3.1). Overlapping segments are then extracted from protein peptides satisfying all the requirements.

## 3.5 Clustering Algorithm

The k-means algorithm [18, 19] is the ideal method for clustering protein segments due to the large input volume. Unfortunately, there are several issues that must first be resolved. The foremost is the requirement to specify the number of clusters k in advance. Some studies have suggested that the numbers of sequence-structure motif clusters are in the hundreds, while others have suggested numbers in the thousands or even as much as over 100K. The wide range makes estimating the number of clusters a groundless act. Moreover, aside from knowing that a larger k generally results in finer clusters, there is not a precise correlation between k and the degree of segment similarity in each cluster. Finally, the original k-means algorithm would fit every segment into its closest cluster, even if that cluster is really nowhere near the segment at all. This would end up contaminating the resultant sequence-structure motif clusters, making them less representative and degrading their capacity to recognize homologous sequence patterns.

The novel clustering algorithm, outlined in Figure 3.3, is intended to solve the aforementioned problems. It is derived from the k-means algorithm and modified to allow a variable number of clusters [48]. An input to the algorithm is the cluster radius *r*, such that a segment either goes to its closest cluster if the distance is within *r*, or starts a new cluster otherwise. The input *r* eliminates the need to estimate and fix the number of clusters, allows a more direct control over the cluster quality, and prevents segments from being forcibly added to faraway clusters. The algorithm also uses a special cluster called the *residue cluster* to hold all outliner segments that cannot be clustered due to their unique sequence patterns or shapes. Since segments in the residue cluster are considered outliners, they are prohibited from initiating new clusters in subsequent iterations. This measure has led to significant runtime improvement as it effectively prevents the creation of tiny miscellaneous clusters.

**Protein Segment Clustering Algorithm**

```
Input: cluster radius $r$, minimum size $m$, segment set $S$, maximum trial count $t$
Create empty residue cluster $C_{res}$
Repeat until no changes or $t$ trials have been exhausted
    For each segment $s \in S$ do
        Find cluster closest to $s$, or set distance to $\infty$ if none exists yet
        If distance $\leq r$ then move $s$ to new cluster and update old cluster
        Otherwise, if $s \notin C_{res}$ then create new cluster with $s$ as centroid
    Merge all nearby clusters (with distance $< 0.5r$)
    For each cluster smaller than $m$ do
        Eliminate cluster and transfer all its segments to $C_{res}$
Return the final set of clusters
```

Figure 3.3: Outline of algorithm for clustering protein segments

## 3.6 Experiments and Results

The algorithm presented in Section 3.5 has been applied to clustering a set of 396 non-redundant proteins selected by Cuff and Barton (CB396) [49]. Segment length $L$ was set to 8, a value small enough to allow clusters of reasonable size but large enough to capture local residue interactions. Results reported by Bystroff et al. [3] have shown that segments of length 8 are very effective at preserving local sequence-dependent information. The cluster radius was set to 1.2 based on the method described in Section 3.3. Both the minimum cluster size and maximum trial count were set to 5. Symbol $\theta$ in Equation (3.1) was set to $120^{\circ}$, an arbitrary but reasonable choice for segments of length 8. A total of 47,907 overlapping segments were extracted from qualified protein peptides (see Section 3.4).

The output consisted of 357 clusters, but the number of distinct structural motifs was much less since many clusters either had the same fold, or were overlapping images of the same motif. For instance, 89 clusters were helices, showing the motif's abundance and its variety in sequence patterns. In short, all motifs in the I-sites library [20, 50] had been discovered together with some new ones. Four examples of new motifs are shown in Figure 3.4 and Figure 3.5. More examples are shown in Appendix A. For each motif, the following information is displayed:

*Segment count* This is the size of the cluster capturing the motif. This number might seem small because it only includes segments that are highly similar to the motif in terms of both sequence and structure.

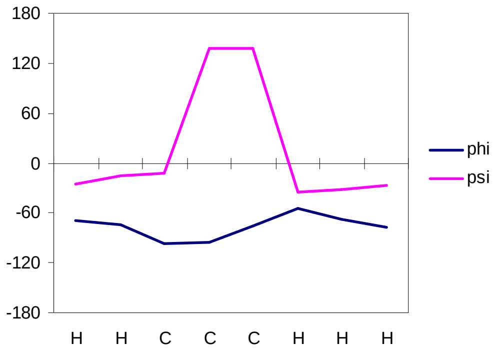
*Dihedral angle plot* The plot shows the *phi* and *psi* angles for each position along the motif and facilitates comparison between structures of different motifs.

*Log-odds profile* For position $i$ and amino acid $j$, entry $v_{ij}$ in the log-odds profile is calculated from $f_{ij}$ (the corresponding entry in frequency profile) and $b_j$ (the background frequency for amino acid $j$) as follows: $v_{ij} = \log_2(f_{ij} / b_j)$.

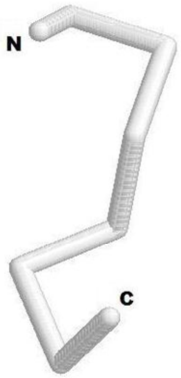
*Backbone drawing* 3D drawing of backbone conformation using Protein Explorer, where the N-terminus is labeled 'N' and the C-terminus is labeled 'C'.

The motif in Figure 3.4(a) represents a turn between two helices, characterized by a MET at position 2, a preference for hydrophobic residues at position 3, and an aversion to them at position 4. In general, positions prior to and including position 3 tend to prefer hydrophobic residues while the others tend prefer hydrophilic ones, inferring a possible emergence from the protein interior to the surface. The motif in Figure 3.4(b) is also a turn flanked by helices. It is characterized by a GLY at position 3, a conserved hydrophobic residue at position 4, and finally an ASX (i.e. ASN or ASP) at position 5. Hydrophilic residues are generally preferred throughout the motif, potentially suggesting that the entire motif is exposed to the aqueous surrounding.

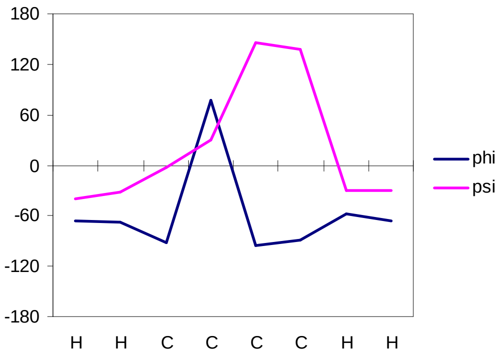
(a) Found in 44 segments



|   | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| **A** | · | 1 | · | · | -1 | · | · | · |
| **R** | -1 | 1 | · | -1 | -1 | · | -1 | · |
| **N** | -1 | · | · | · | 1 | · | · | · |
| **D** | -1 | 1 | · | · | 2 | · | 1 | 1 |
| **C** | · | -3 | · | · | -1 | -3 | -2 | · |
| **Q** | · | · | · | · | · | · | 1 | 1 |
| **E** | · | 1 | · | -1 | · | · | 1 | 1 |
| **G** | -1 | -1 | -1 | -1 | · | -1 | -1 | -1 |
| **H** | -2 | · | · | · | · | -1 | -1 | · |
| **I** | · | -1 | · | 1 | -2 | · | -2 | -2 |
| **L** | 1 | -1 | · | 1 | -1 | · | -1 | -1 |
| **K** | · | 1 | · | -1 | · | · | · | · |
| **M** | 1 | · | 1 | 1 | -1 | · | -2 | -1 |
| **F** | · | -1 | · | · | -2 | -1 | -2 | -1 |
| **P** | · | · | -2 | -2 | 2 | 1 | · | -1 |
| **S** | · | · | · | · | 1 | · | · | · |
| **T** | · | · | · | · | · | · | · | 1 |
| **W** | 1 | -2 | · | -1 | -2 | -1 | -2 | · |
| **Y** | · | -1 | · | · | -1 | · | -1 | · |
| **V** | · | -1 | · | 1 | -1 | -1 | -1 | · |



(b) Found in 36 segments



|   | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| **A** | · | 1 | 1 | -1 | -1 | -1 | · | · |
| **R** | 1 | · | · | -1 | · | -1 | · | 1 |
| **N** | · | · | · | 1 | -2 | 1 | · | · |
| **D** | · | · | -1 | · | -2 | 2 | -1 | 1 |
| **C** | -2 | -2 | -2 | -2 | · | -2 | -1 | -1 |
| **Q** | · | 1 | · | · | · | -2 | -1 | · |
| **E** | 1 | 1 | · | -1 | -1 | · | · | 1 |
| **G** | -2 | -1 | -1 | 3 | -2 | · | -1 | -1 |
| **H** | · | · | 1 | -1 | -1 | -1 | · | · |
| **I** | -1 | -1 | -1 | -3 | 1 | -3 | · | -2 |
| **L** | -1 | -1 | 1 | -2 | 1 | -2 | 1 | -1 |
| **K** | 2 | 1 | · | · | · | · | · | 1 |
| **M** | -1 | -1 | 1 | -1 | 1 | -2 | · | -1 |
| **F** | -2 | -2 | 1 | -2 | 1 | -1 | 1 | -2 |
| **P** | -2 | -2 | -4 | -2 | · | 1 | 2 | · |
| **S** | · | · | -1 | -1 | -2 | 1 | -1 | · |
| **T** | -1 | · | · | -3 | -1 | 1 | -1 | -1 |
| **W** | -1 | · | -1 | -1 | 1 | -1 | · | -1 |
| **Y** | -1 | -2 | 1 | -2 | 1 | -1 | · | -1 |
| **V** | -1 | -1 | -2 | -3 | 1 | -2 | · | -1 |

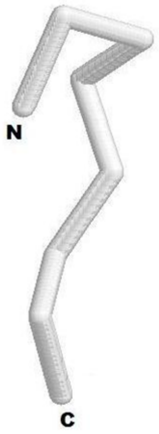
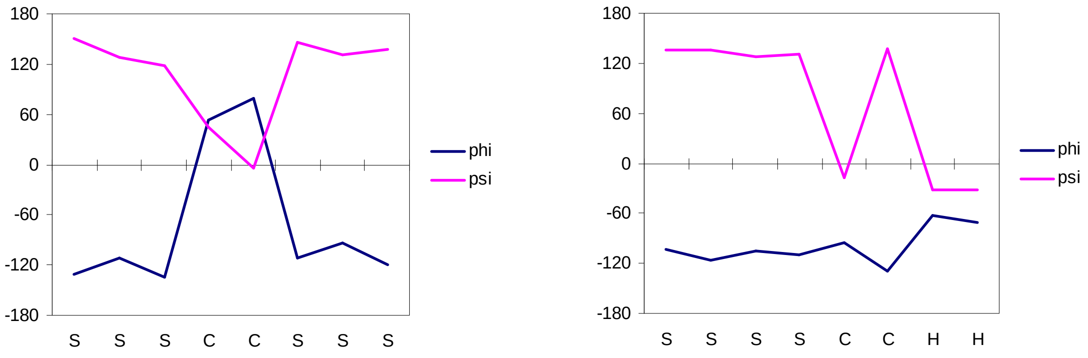


Figure 3.4: Dihedral angles, log-odds profiles, and 3D backbone drawings for two novel motifs not listed in the I-sites Library. Dot (·) represents background frequency.

(a) Found in 25 segments

(b) Found in 21 segments



|   | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| **A** | · | · | -1 | -1 | -1 | · | -1 | -1 |
| **R** | · | · | · | · | · | 1 | -1 | 1 |
| **N** | -1 | · | · | 2 | 1 | 1 | -1 | -1 |
| **D** | -2 | 1 | 1 | 2 | 1 | · | -1 | -1 |
| **C** | · | -1 | -1 | -4 | -3 | -3 | -1 | 1 |
| **Q** | · | 1 | · | · | -1 | 1 | · | · |
| **E** | -1 | · | · | · | · | 1 | · | · |
| **G** | -2 | -1 | · | 1 | 2 | · | -1 | -2 |
| **H** | -1 | · | · | -1 | -1 | · | -2 | · |
| **I** | 1 | · | · | -3 | -2 | -2 | 1 | 1 |
| **L** | · | · | · | -3 | -2 | -2 | 1 | · |
| **K** | -1 | · | · | 1 | · | 2 | · | · |
| **M** | · | · | · | -2 | -3 | -1 | · | -1 |
| **F** | 1 | · | · | -2 | -2 | -1 | · | 1 |
| **P** | -1 | -2 | -1 | -1 | -1 | -1 | · | -2 |
| **S** | -1 | · | · | · | · | · | -1 | -1 |
| **T** | 1 | · | · | -1 | -1 | · | 1 | · |
| **W** | -2 | 1 | 2 | -4 | -2 | -2 | 1 | -1 |
| **Y** | 1 | · | 1 | -2 | -1 | · | · | 1 |
| **V** | 1 | 1 | · | -3 | -1 | · | 1 | 1 |

|   | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| **A** | · | · | · | · | · | -1 | · | · |
| **R** | -1 | · | · | -2 | · | -1 | · | · |
| **N** | -1 | -1 | · | -2 | · | 2 | -1 | 1 |
| **D** | -1 | -1 | -1 | -2 | 1 | 2 | · | 1 |
| **C** | · | -1 | -2 | · | -3 | -2 | -2 | -3 |
| **Q** | -1 | 1 | -1 | -1 | -1 | -1 | · | · |
| **E** | · | · | · | -2 | 1 | · | · | 2 |
| **G** | -1 | -2 | -1 | -2 | -1 | · | -1 | -1 |
| **H** | · | 1 | -1 | · | · | · | -1 | -1 |
| **I** | · | 1 | · | 1 | -1 | -3 | · | -2 |
| **L** | · | 1 | · | · | · | -2 | · | -1 |
| **K** | · | -1 | · | -2 | 1 | -1 | 1 | · |
| **M** | -1 | -1 | · | -1 | -1 | -3 | -1 | -2 |
| **F** | 1 | · | -1 | · | -1 | -2 | -1 | -1 |
| **P** | · | -1 | 1 | -1 | 1 | -1 | 1 | -1 |
| **S** | -1 | -1 | · | -1 | · | 1 | · | · |
| **T** | · | -1 | 1 | · | · | · | · | · |
| **W** | -1 | -1 | · | · | · | -2 | · | -1 |
| **Y** | 1 | 1 | · | · | · | -2 | -1 | -2 |
| **V** | 1 | 1 | 1 | 2 | · | -2 | · | -2 |

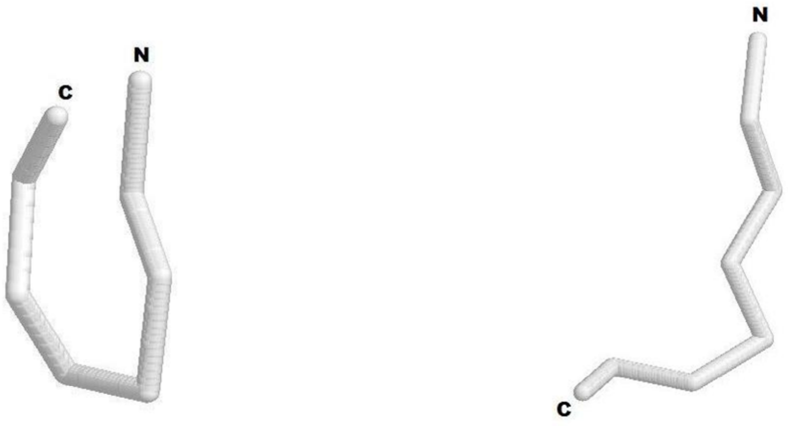


Figure 3.5: Dihedral angles, log-odds profiles, and 3D backbone drawings for two other novel motifs not listed in the I-sites Library. Dot (·) represents background frequency.

On the other hand, the motif in Figure 3.5(a) is very similar to the PDG beta-hairpin listed in the I-sites library. While both motifs possess the conserved sequence ASX-GLY, the PDG hairpin has an additional conserved PRO prefixing the sequence. Similar to the PDG hairpin, this motif also forms a hairpin by having two anti-parallel strands connected by a U-shaped turn. Finally, the motif in Figure 3.5(b) represents a turn linking a strand and a helix. It is characterized by a conserved hydrophilic residue at position 4 and an ASX at position 5. The preceding strand positions are mostly hydrophobic, indicating that the motif is likely to protrude from the protein interior.

These examples illustrate the competency of the clustering method at discovering local protein motifs, revealing their unique compositions, and identifying their relative locations within proteins. Note that it is difficult to conduct a fair comparison between clustering methods due to the vastly different settings. Nevertheless, given that the novel method was able to discover all motifs in the I-sites Library and more, one may conclude that it is comparable, if not better, than the method of Bystroff and Baker [20].

# Chapter 4

# Local Tertiary Structure Prediction

The first application of sequence-structure motif clusters is aimed at the prediction of local tertiary structures based on sequence composition alone, and a novel algorithm based on dynamic programming (DP) has been invented for that purpose. This section begins with the definition of *cluster assignment* and *assignment rank*, two important concepts appearing in the algorithm. It then describes the two preprocessing steps taken to improve the prediction capacity of a given cluster set, namely the removal of *noise clusters* and the enhancement to the cluster assignment scoring function. Finally, the prediction algorithm is covered in detail, and performance results gathered from a comprehensive experiment are presented.

## 4.1 Cluster Assignment and Assignment Rank

Scoring function $K_c(s)$, shown in Equation (4.1), computes the likelihood of a length-$L$ segment $s$ belonging to cluster $c$ based on sequence composition. It is derived from the log-odds ratio of the probability of observing $s$ given $c$ to the background probability. Symbols $s_{ij}$ and $c_{ij}$ denote the frequency of amino acid *j* at position *i* on *s* and *c*'s centroid respectively. Symbol $b_j$ denotes the background frequency for amino acid *j*.

$$K_c(s) = \log_2 \left( \frac{\prod_{i=0}^{L-1} \sum_{j=0}^{19} s_{ij}\, c_{ij}}{\prod_{i=0}^{L-1} \sum_{j=0}^{19} s_{ij}\, b_j} \right) \tag{4.1}$$

The main purpose of the scoring function is for making *cluster assignments*. Another related concept that is equally important is the *assignment rank*. Both concepts are central to the algorithm for predicting local tertiary structure (Section 4.5) and enhancing secondary structure prediction (Section 5.1). Their definitions are as follows.

**Definition 4.1.1 (Cluster assignment).** A *cluster assignment*, or just *assignment*, refers to an instance when a cluster is assigned to a segment based on a score computed via Equation (4.1). The assignment is said to cover the segment and its residues. Each assignment has three basic attributes: the cluster being assigned, the segment being covered, and the score associated with the pair.

**Definition 4.1.2 (Assignment rank).** An algorithm utilizing an *assignment rank*, or just *rank*, of *R* means that the *R* highest scoring assignments are made to each segment for the task at hand. The highest scoring assignment is at rank 1, the second highest at rank 2, and so on.

## 4.2 Evaluation of Local Structure Prediction

The evaluation scheme for local tertiary structure prediction was invented by Lesk [51]. It takes two parameters, a window size *w* and a RMSD threshold *t*. Given a true structure and its prediction, the scheme computes the percentage of residues found in length-*w* segments whose predicted structures are within *t* of the true structure after superposition [52]. The parameters used by Bystroff and Baker [20] are selected to facilitate comparison (i.e. $w = 8$ and $t = 1.4$ Å).

## 4.3 Noise Cluster Elimination

In a large cluster set, some weak clusters capturing rare motifs possess similar sequence profiles as do the significant clusters capturing more common motifs. Those weak clusters tend to compete with the significant clusters for sequence similarity with target segments during cluster assignment, degrading prediction accuracy. Because they create noise that disturbs prediction, those weak clusters are called *noise clusters* and should be eliminated.

Clusters produced by the algorithm described in Section 3.5 are of minimum size *m*. If *m* is set too small, many noise clusters arise. If it is set too large, significant clusters are lost. To determine *m* maximizing the predictive power for a set of clusters, the following method is used.

**Noise Cluster Elimination**

```
Input: cluster set $C$, protein set $P$, minimum size bound $[m_l, m_h]$
For each $m$ in range $[m_l, m_h]$
    Remove clusters of size less than $m$ from $C$ to obtain $C'$
    Get average prediction accuracy for $P$ using $C'$ as follows:
        For each protein $p \in P$ do
            Assign highest scoring cluster to each overlapping segment in $p$
            Sort all assignments by score
            Assign structures to $p$ from highest scoring assignments
            Evaluate prediction as described in Section 4.2
Return $m$ and $C'$ resulting in highest average prediction accuracy
```

Figure 4.1: Outline of procedure for eliminating noise clusters

Figure 4.2 shows the fluctuation in prediction accuracy as *m* increased from 5 to 25 inclusive. While the prediction accuracy remained rather constant in the middle stretch, it rose and fell sharply at both ends. Prediction was compromised by the presence of noise clusters for small *m* (< 8) and the absence of significant clusters for large *m* (> 20). The optimal minimum cluster size was *m* = 16, yielding a prediction accuracy of 54.66%.

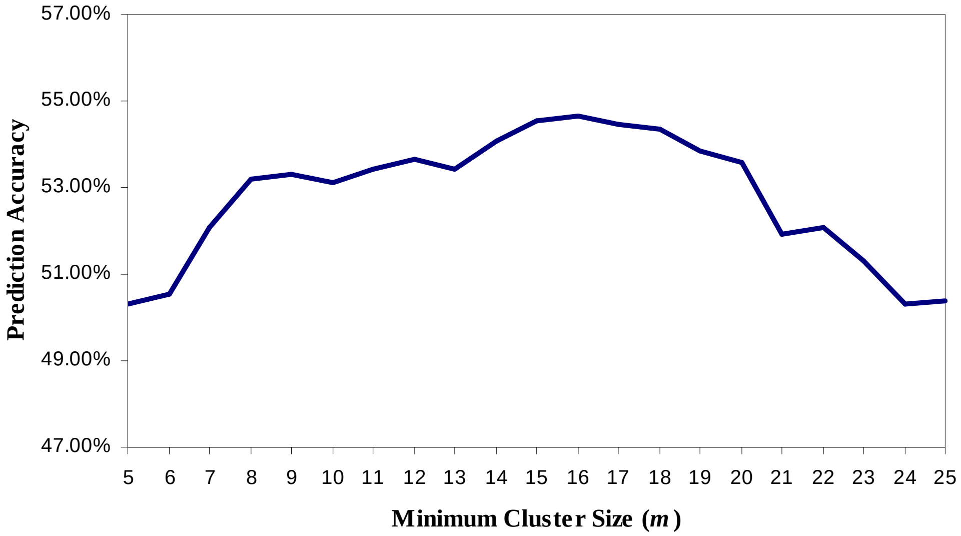


Figure 4.2: Fluctuation in prediction accuracy as minimum cluster size *m* rises from 5 to 25

## 4.4 An Enhanced Cluster Likelihood Function

As described in Section 4.1, cluster assignments are made based on similarity scores computed via the likelihood function shown in Equation (4.1). This section improves the function with the addition of a new term, as shown in Equation (4.2) below.

$$K_c(s) = \log_2 \left( \frac{\prod_{i=0}^{L-1} \sum_{j=0}^{19} s_{ij} \, c_{ij}}{\prod_{i=0}^{L-1} \sum_{j=0}^{19} s_{ij} \, b_j} \right) - base_c \qquad (4.2)$$

The new term $base_c$ represents the *cluster-specific base cutoff* for cluster $c$. Equation (4.1) simply assumes a cutoff of 0 for all clusters, an intuitive choice for log-odds. The derivation of cluster-specific base cutoffs is based on a simple observation. The rarer the motif a cluster represents, the more likely that segments classified to the cluster are false positives ($F^+$), and the higher the cutoff has to be raised to avoid a high $F^+$ rate. In contrast, if a cluster represents a common motif, then segments not classified to it are likely to be false negatives ($F^-$), so the cutoff has to be lowered to suppress the $F^-$ rate. The derivation procedure for the cutoffs is shown in Figure 4.3, where *sign*($x$) returns 1 if $x \geq 0$ or -1 otherwise.

**Derivation of Cluster-Specific Base Cutoff**

```
Input: cluster set $C$, protein segment set $S$, small positive value $\varepsilon$
For each segment $c \in C$ do
    $T^+ = \{s \in S \mid s$ is most likely to belong to $c$ based on (3) AND
                  $s$ and $c$ share similar structures$\}$
    $F^+ = \{s \in S \mid s$ is most likely to belong to $c$ based on (3) AND
                  $s$ and $c$ have different structures$\}$
    $base_c = 0$
    While $f^+$ and $f^-$ are not sufficiently close do
        $f^-$ = # segments in $T^+$ with likelihood score from (3) $< base_c$
        $f^+$ = # segments in $F^+$ with likelihood score from (3) $\geq base_c$
        $base_c = base_c + sign(f^+ - f^-) * \varepsilon$
Return the set of $base_c \, \forall \, c \in C$
```

Figure 4.3: Outline of derivation procedure for cluster-specific base cutoffs

Recall from Figure 4.2 that the highest accuracy reached was 54.66% for $m = 16$. Once switched to Equation (4.2), the accuracy climbed to 56.7%. Note that the forthcoming definition is to override Definition 4.1.1 for the remainder of this thesis. The only difference is that Definition 4.1.1 refers to Equation (4.1) while the new definition refers to Equation (4.2).

**Definition 4.4.1 (Cluster assignment).** A *cluster assignment*, or just *assignment*, refers to an instance when a cluster is assigned to a segment based on a score computed via Equation (4.2). The assignment is said to cover the segment and its residues. Each assignment has three basic attributes: the cluster being assigned, the segment being covered, and the score associated with the pair.

## 4.5 Local Structure Prediction using Dynamic Programming

Let $R$ be the assignment rank, $L$ be the segment length, and $p$ be the target protein of length $n$. The initial setup for the algorithm involves making the $R$ highest scoring cluster assignments to each overlapping length-$L$ segment along $p$. Let $a_{ir}$ denote the assignment at rank $r$ starting at position $i$, where $1 \leq r \leq R$ and $0 \leq i \leq n–L$. Define $A_i = \{a_{ir} \forall r\}$ and $A = \{a_{ir}\}$. The set $A$, depicted in Figure 4.4(a), forms the entire search space for the algorithm.

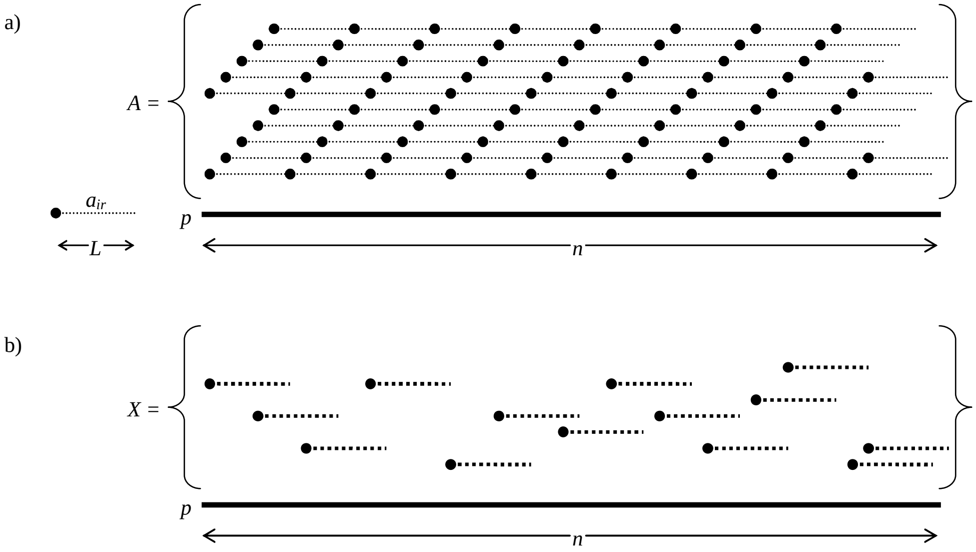


Figure 4.4: (a) Assignment set $A$ consists of all individual assignments $a_{ir}$ of length $L$ covering target protein $p$ of length $n$. Assignment rank $R$ is 2, the number of assignments made to each overlapping length-$L$ segment in $p$. Each assignment $a_{ir}$, represented as a big dot (●) with a dotted tail, covers residues $i$ to $i$+$L$–1 inclusive. (b) $X$ is a subset of $A$ that covers all residues in $p$, formed by linking adjoining assignments together.

The goal is to compute a subset $X^* \subseteq A$ such that $X^*$ covers all residues in $p$ and maximizes a certain objective function. An example of a legitimate candidate subset $X$ is shown in Figure 4.4(b). The objective function is derived in light of two observations. First, the cluster assignment most appropriately capturing the shape of a segment might not always be the optimal (i.e. highest scoring) one but a sub-optimal one. Second, if overlapping assignments have serious structural conflicts among themselves, then they should not be adopted together. Having taken both factors into consideration, Equation (4.3) is proposed as the objective function for measuring the quality of an assignment set $X$ when used to form a prediction for a protein of length $n$.

$$F(X) = q\sum_{i=0}^{n-1} score(X, i) - \sum_{i=0}^{n-1} conflict(X, i) \tag{4.3}$$

Function $F(X)$ returns the objective score for assignment set $X$. Symbol $q$ is a non-negative constant for balancing the two parts representing the total score and conflict induced by $X$. It is set to 70 in this study, a value found empirically to yield one of the best predictions. Functions *score*($X$, $i$) and *conflict*($X$, $i$) are defined in Equation (4.4) and Equation (4.5) respectively.

$$score(X,i) = \begin{cases} \text{score of highest scoring assignment in } X \text{ covering residue } i \text{, or} \\ 0 \text{ if no assignment in } X \text{ covers residue } i \end{cases} \tag{4.4}$$

$$conflict(X,i) = \begin{cases} \overline{\Delta\varphi} + \overline{\Delta\psi} \text{ between all pairs of assignments in } X \text{ at positions} \\ \quad \text{covering residue } i \text{, or} \\ 0 \text{ if at most 1 assignment in } X \text{ covers residue } i \end{cases} \tag{4.5}$$

Symbols $\overline{\Delta\varphi}$ and $\overline{\Delta\psi}$ denote the mean absolute difference in *phi* and *psi* angles respectively. Now, the algorithm is to take a dynamic programming (DP) approach to compute the assignment set $X^*$ that covers all residues in $p$ and is optimal (i.e. maximizing objective function $F$).

Assignment sets are built starting from the head of $p$ by appending or concatenating to the end one adjoining assignment at a time. Note that simply extending the current optimal set by adding to its tail the best available adjoining assignment does not guarantee optimality for the resultant set. The assignment just added may overlap with existing assignments in the set, introducing new conflicts that must be fixed by replacing those assignments, which in turn may cause more new conflicts with their prior overlapping assignments and necessitate further replacements. To avoid such propagation of conflict, a more involved DP algorithm is needed.

When any assignment $\alpha \in A_i$ is appended to the end of assignment set $X$, it would come in contact with one or more trailing assignments in $X$. The relative arrangement of these trailing assignments and their ranks collectively form the *tail configuration* for $X$ with respect to $\alpha \in A_i$, denoted by $tail_i(X)$. Note that $tail_i(X)$ is defined to be an *empty tail configuration* if $X$ is too short to reach any assignment in $A_i$. For formulation purposes, $tail_j(X)$ is allowed for $j > n–L$ as if $A_j$ actually existed. Figure 4.5 shows the set of all possible non-empty tail configurations for $L = 3$ and $R = 1$ with respect to $\alpha$, the assignment to be appended.

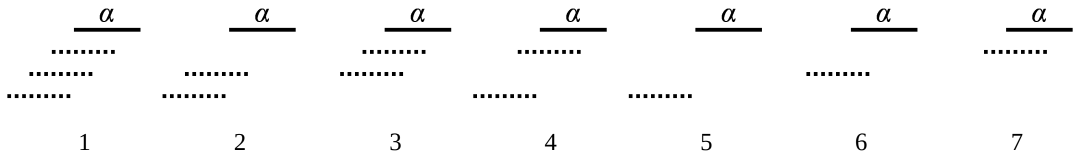


Figure 4.5: All seven unique non-empty tail configurations for $L = 3$ and $R = 1$. Each line denotes an assignment. In each case, $\alpha$ (solid line) is the assignment to be appended to a set $X$, and the set of all trailing assignments in $X$ touched by $\alpha$ (dotted lines) forms the tail configuration w.r.t. $\alpha$.

For each position $i$ starting from the head of $p$, the algorithm computes $V_i$, the set of all optimal assignment sets $X$ with unique non-empty $tail_{i+1}(X)$. The DP recurrence for the algorithm is stated in Figure 4.6.

**DP Recurrence for Local Tertiary Structure Prediction**

***Initial condition:***

$V_0 = \{\{\alpha\}\ \forall\ \alpha \in A_0\}$

***Inductive hypothesis for position i, 0 ≤ i ≤ n – L:***

$V_i = \{$All optimal assignment sets $X$ with unique non-empty $tail_{i+1}(X)\}$

***Recurrence:***

Let $V_i' = \{W \cup \{\alpha\}\ \forall\ W \in V_i$ and $\alpha \in A_{i+1}\}$

For each unique non-empty tail configuration $t'$

$V_{(i+1)t'} = X \in \{W \in V_i \cup V_i' \mid tail_{i+2}(W) = t'\}$ s.t. $F(X)$ is maximized

Let $V_{i+1} = \{V_{(i+1)t'}\}$

***Final solution:***

$X^* = X \in \{W \in V_{n-L} \mid W$ has an assignment in $A_{n-L}\}$ s.t. $F(X)$ is maximized

Figure 4.6: DP recurrence for local tertiary structure prediction

The recurrence ensures the optimality for each $V_{(i+1)t'}$, and the uniqueness and non-emptiness of the associated tail configuration $t'$, so the inductive hypothesis holds for position $i$+1. Finally, dihedral angles are assigned to the residues in $p$ by back-tracking the creation of $X^*$.

## 4.6 Time Complexity of DP Algorithm

A bound on the size of $V_i$ is required in order to analyze the time complexity of the DP algorithm just described. By definition, $|V_i|$ is at most the total number of all unique non-empty tail configurations. Figure 4.5 lists all seven possible unique non-empty tail configurations for segment length $L = 3$ and assignment rank $R = 1$ with respect to assignment $\alpha$. For general $L$ and $R$, note that when $\alpha$ is appended to an assignment set $X$, it could be touching anywhere from 1 to $L$ trailing assignments in $X$, each of which is selected from a pool of size $R$. Further, the $k$ trailing assignments being touched could be any $k$ out of a total of $L$. Let $T_k$ represent the number of unique non-empty tail configurations comprising $k$ assignments. It can be computed as follows according to basic counting principles:

$$T_k = \binom{L}{k} R^k \tag{4.6}$$

Summing all $T_k$ gives the total number of unique non-empty tail configurations $T$:

$$T = \sum_{k=1}^{L} T_k = \sum_{k=1}^{L} \binom{L}{k} R^k = \sum_{k=0}^{L} \binom{L}{k} R^k \, 1^{L\text{-}k} \text{ -1} = (R+1)^L \text{ -1} \tag{4.7}$$

Consequently, the bound $|V_i| \leq T = (R+1)^L – 1$ holds. For each position $i$, the algorithm calculates the objective value for $|V_i|$ * $R$ new assignment sets, where each calculation takes $O(L^3)$ if done carefully. Hence, the total runtime is $O(n\ |V_i|\ R\ L^3) = O(n\ L^3\ (R+1)^{L+1})$ for all $n$ positions. Despite the exponential term, typical values for $R$ and $L$ are small enough to make the algorithm feasible (e.g. $R = 3$ and $L = 8$ in this study).

## 4.7 Experiments and Results

### 4.7.1 Rotation Test

The four protein sets used for training and testing in this test were the testing set of 55 proteins used by Bystroff and Baker (BB55) [20], the training set of 126 proteins introduced by Rost and Sander (RS126) [37], the testing set of 187 proteins for PSIPRED (PP187) [14], and finally the testing set of 396 proteins selected by Cuff and Barton (CB396) [49]. A listing of the proteins in each data set can be found in Appendix B.

The test method started by picking one set to be the training set for cluster creation and using the rest as testing sets. After gathering results, the method rotated the sets such that a different set became the training set and the others became testing sets. The method continued until all sets had been used for training. Doing such rotation helped avoid biased results due to dataset-dependency and test data insufficiency. Assignment rank *R* was set to 3 throughout the test. The results are summarized in Table 4.1.

Table 4.1: Prediction accuracy of the DP algorithm obtained from the rotation test, evaluated using the scheme described in Section 4.2. "Min. Size" refers to the optimal minimum size as described in Section 4.3, and "# Clusters" refers to the number of clusters with at least the minimum size.

| **Training Set** | **Min. Size / # Clusters** | **Testing Set** | **Helix Accuracy** | **Strand Accuracy** | **Coil Accuracy** | **Overall Accuracy** | **Average** |
|---|---|---|---|---|---|---|---|
| BB55 | 7 / 40 | RS126 | 81.39% | 55.05% | 43.25% | 58.35% | 58.00% |
| | | PP187 | 78.56% | 51.71% | 41.61% | 56.86% | |
| | | CB396 | 80.26% | 52.95% | 43.21% | 58.78% | |
| RS126 | 10 / 58 | BB55 | 79.37% | 54.01% | 40.61% | 57.74% | 58.32% |
| | | PP187 | 80.80% | 52.43% | 41.61% | 57.81% | |
| | | CB396 | 82.52% | 53.03% | 42.75% | 59.42% | |
| PP187 | 7 / 161 | BB55 | 79.26% | 49.18% | 41.83% | 57.22% | 58.52% |
| | | RS126 | 83.95% | 51.75% | 43.46% | 58.49% | |
| | | CB396 | 83.28% | 50.47% | 44.51% | 59.84% | |
| CB396 | 16 / 164 | BB55 | 84.58% | 46.31% | 43.71% | 59.39% | 59.88% |
| | | RS126 | 87.69% | 48.05% | 45.41% | 59.69% | |
| | | PP187 | 87.17% | 48.56% | 44.96% | 60.55% | |
| Average | | | 82.40% | 51.13% | 43.08% | 58.68% | |

### 4.7.2 Jackknife Test

Since the data sets used in the rotation test (i.e. PP55, RS126, PP187, and CB396) were selected independently, members in different sets might be highly similar or even identical. Such overlaps could have inflated the prediction accuracy and thus prevented the rotation test from impartially evaluating the performance of the algorithm. As a remedy, a jackknife test ensuring absolutely no overlaps between training and testing sets was conducted.

The jackknife test was performed on CB396 [49], a set of 396 peptides selected through a very stringent procedure to ensure non-redundancy between members. The entire test consisted of 10 iterations, each of which involved splitting CB396 into two disjoint subsets in 80/20 ratio by residue count. The larger subset was then used for training and the smaller one for testing.

Note that testing sets containing more helices tend to yield higher accuracies than those containing more coils. Consequently, for results to be consistent, all testing sets should contain similar proportions of each secondary structure (SS). To guarantee such condition, the background proportion of each SS was first estimated from the whole CB396. Each repetition of the jackknife test then produced 50 pairs of training and testing sets, and used the pair whose testing set exhibited SS proportions most closely resembling the background ones. Table 4.2 shows the results from the jackknife test, using the same assignment rank as the rotation test (i.e. $R$ = 3).

Table 4.2: Prediction accuracy of the DP algorithm obtained from a ten-iteration jackknife test and evaluated using the scheme described in Section 4.2.

| **Jackknife Test Iteration #** | **Helix Accuracy** | **Strand Accuracy** | **Coil Accuracy** | **Overall Accuracy** |
|---|---|---|---|---|
| 1 | 86.22% | 44.91% | 40.58% | 58.51% |
| 2 | 84.54% | 39.61% | 40.52% | 56.19% |
| 3 | 84.27% | 44.71% | 41.07% | 58.03% |
| 4 | 84.65% | 43.22% | 40.19% | 57.45% |
| 5 | 86.12% | 43.63% | 42.87% | 59.15% |
| 6 | 88.11% | 42.92% | 43.05% | 59.69% |
| 7 | 84.83% | 44.35% | 40.99% | 57.76% |
| 8 | 84.94% | 43.98% | 42.90% | 58.83% |
| 9 | 86.09% | 43.22% | 42.78% | 58.86% |
| 10 | 83.68% | 45.30% | 40.61% | 57.62% |
| Average | 85.35% | 43.59% | 41.56% | 58.21% |

Average accuracy obtained in the rotation test (i.e. 58.68%) is higher than that obtained in the jackknife test (i.e. 58.21%). If overlaps between data sets were responsible for the slight difference of 0.47%, then this test would confirm the negligibility of the overlaps and uphold the validity of the results in the rotation test. Additionally, this test has also illustrated the consistent performance of the algorithm as similar accuracies were observed across all iterations.

### 4.7.3 Discussion

Both tests have shown that over 58% of all residues on average were found in at least one length-8 segment whose predicted structure was within 1.4 Å of the true structure, measured in RMSD. This is significant considering that the prediction relied solely on sequence information, without taking into account global forces such as disulfide bridges, hydrophobic effects, inter-group charges, and so on. The result is also a great improvement over that published by Bystroff and Baker [20], which was 50% (see Section 2.2.2). Although the method of Wang and Yang [21] produced better numerical results, it used over 100K motif clusters and yielded only approximate predictions (see Section 2.2.3). The algorithm described here, for example, used at most 164 clusters in the rotation test and produced predictions with precise backbone conformations. Taking all the factors into consideration, both methods would be very much comparable.

While all four training sets yielded similar results according to Table 4.1, a general trend existed in which the more clusters the training involved, the higher the average accuracy reached. Besides overlaps between data sets, which have been deemed insubstantial by the jackknife test, another possible reason would be that a larger cluster set constituted a larger conformational search space and consequently contributed to better predictions. The real surprising observation, however, is that the number of clusters had only minimal effects on the prediction accuracy. For instance, using a set of 40 clusters (created from BB55) yielded 58% accuracy, while using another with 164 clusters (created from CB396) yielded 60% accuracy. Although the difference of nearly 2% was significant, one might have expected more given the large deviation in cluster counts. The likely explanation is that the larger clusters were already sufficient to account for the common structures in the test proteins, leaving the smaller clusters to handle only the rarer shapes. This in turn confirms the effectiveness of the clustering method described in Chapter 3, as the larger clusters produced were indeed able to capture the majority of protein conformations.

A breakdown in overall prediction accuracy in both tests by secondary structure states reveals the real strengths and weaknesses of prediction using clusters. Helices were by far the most accurately predicted because they were the most conserved and abundant local motifs. Strands, albeit well conserved, were a lot harder to predict as their formation involved long-range residue interactions, something not captured by local motif clusters. Coils were the most difficult to predict since most of them lacked virtually any kind of detectable conserved patterns.

# Chapter 5

# Secondary Structure Prediction

The second application of sequence-structure motif clusters deals with enhancing secondary structure (SS) prediction. The target predictor [39] is the one based on Support Vector Machines (SVM) [42], so selected because it is one of the best available. As an overview, the procedure involves building a *Secondary Structure Confidence Profile* (*SSCP*) and using it as additional data for training and classification.

## 5.1 Secondary Structure Confidence Profile (SSCP)

The SSCP of a protein shows the confidence, or probability, of each residue being in each of the three SS states, namely helix (H), strand (E), and coil (C). Figure 5.1 shows the SSCP for a section of the protein identified as 1LCL in PDB.

| *Seq* | P | Y | T | E | A | A | S | L | S | T | G | S | T | V | T | … |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| *Helix* | 0.40 | 0.46 | 0.36 | 0.36 | 0.31 | 0.28 | 0.22 | 0.21 | 0.09 | 0.04 | 0.05 | 0.03 | 0.03 | 0.03 | 0.03 | … |
| *Strand* | 0.27 | 0.32 | 0.29 | 0.31 | 0.31 | 0.33 | 0.38 | 0.30 | 0.14 | 0.10 | 0.11 | 0.20 | 0.84 | 0.89 | 0.90 | … |
| *Coil* | 0.33 | 0.22 | 0.35 | 0.33 | 0.38 | 0.39 | 0.40 | 0.49 | 0.77 | 0.86 | 0.84 | 0.77 | 0.13 | 0.08 | 0.07 | … |

Figure 5.1: Secondary structure confidence profile (SSCP) for a section of protein 1LCL

Let $R$ be the assignment rank and $p$ be the target protein. The procedure for generating SSCP starts by making the $R$ highest scoring assignments to each overlapping segment in $p$. Then, for each residue $i$ and SS label $s \in \{H, E, C\}$, it computes $score_{is}$ by summing the scores of all assignments covering $i$ with label $s$ at the covering position. The value $score_{is}$ is then normalized to obtain $ssc_{is}$, the SS confidence for $i$ belonging to state $s$. That is, $ssc_{is} = score_{is}$ / ($score_{iH} + score_{iE} + score_{iCs}$). The set of all $ssc_{is}$ constitutes the SSCP for $p$.

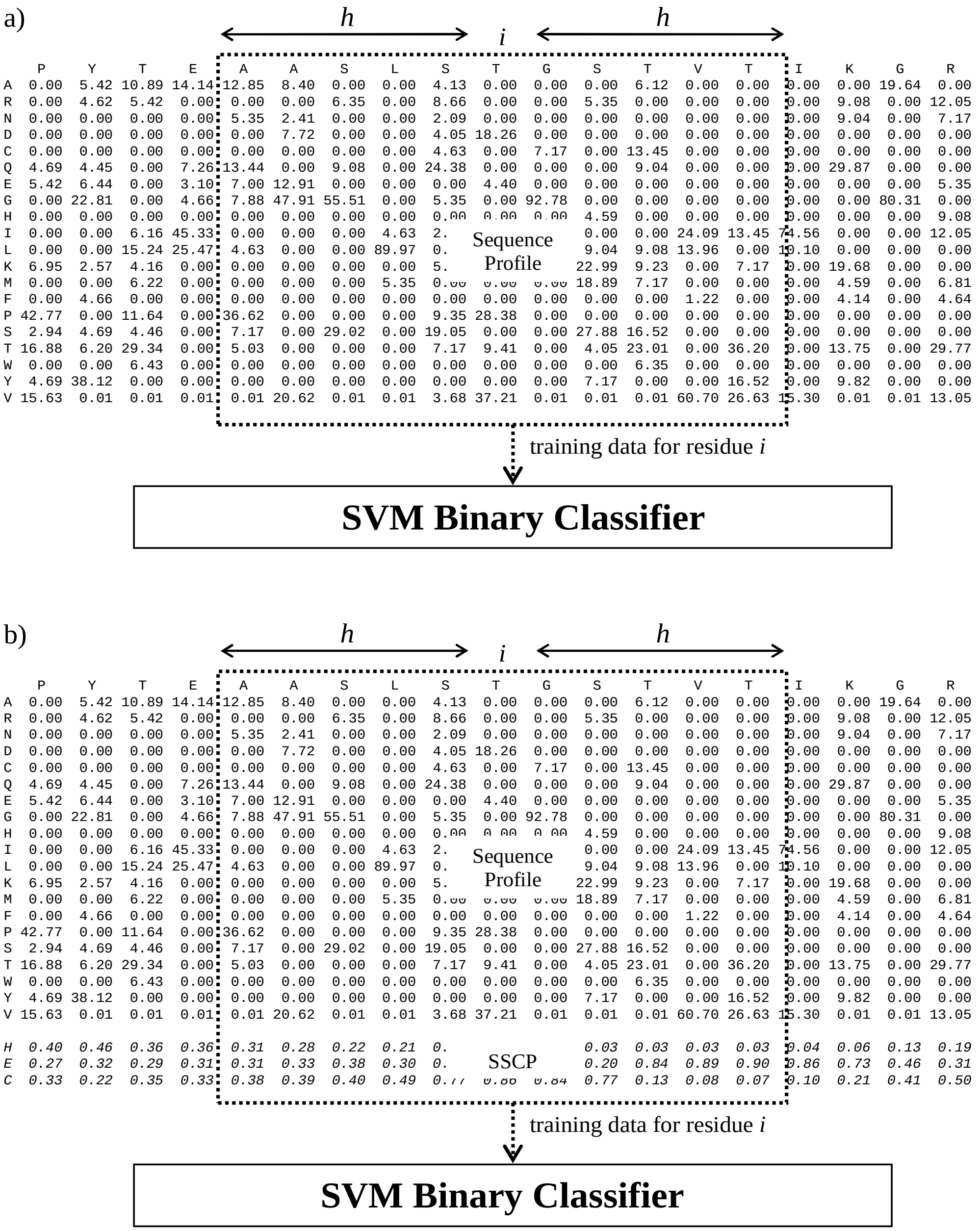
## 5.2 Training of SVM Binary Classifiers

The training procedure is similar to the one used by Hua and Sun [39]. Fix a window half-width $h$ such that each residue is represented by the sequence profile spanning ($2h + 1$) columns, with the said residue in the middle. Each column is coded using 21 entries, where the extra entry is set when the window is extended beyond the ends of a protein [53]. Together, each residue is coded by a total of ($2h + 1$) * 21 entries. When SSCP is incorporated into training, each column is coded with four additional entries. Each of the first three holds the SSCP confidence value for a different SS state, and the last is again set for the case when the window is extended beyond the ends of a protein. Hence, each residue is now coded by a total of ($2h + 1$) * 25 entries. The conceptual view of training with and without SSCP is shown in Figure 5.2.

## 5.3 SVM Predictor Construction

Hua and Sun [39] have demonstrated that the arrangement of SVM binary classifiers has a significant impact on the performance of the resultant SS predictor. This study has adopted an arrangement called *SVM MAX*, one of the most effective arrangements among those Hua and Sun have considered. SVM MAX comprises three SVM binary classifiers, namely H/~H, E/~E, and

C/~C. Each target residue is fed in parallel to all three classifiers, and assigned the SS label corresponding to the one giving the largest decision value. For optimal prediction, the half-width *h* for the three classifiers is set to 5, 4, and 3 respectively.



```
     P     Y     T     E     A     A     S     L     S     T     G     S     T     V     T     I     K     G     R
A  0.00  5.42 10.89 14.14 12.85  8.40  0.00  0.00  4.13  0.00  0.00  0.00  6.12  0.00  0.00  0.00  0.00 19.64  0.00
R  0.00  4.62  5.42  0.00  0.00  0.00  6.35  0.00  8.66  0.00  0.00  5.35  0.00  0.00  0.00  0.00  9.08  0.00 12.05
N  0.00  0.00  0.00  0.00  5.35  2.41  0.00  0.00  2.09  0.00  0.00  0.00  0.00  0.00  0.00  0.00  9.04  0.00  7.17
D  0.00  0.00  0.00  0.00  0.00  7.72  0.00  0.00  4.05 18.26  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00
C  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  4.63  0.00  7.17  0.00 13.45  0.00  0.00  0.00  0.00  0.00  0.00
Q  4.69  4.45  0.00  7.26 13.44  0.00  9.08  0.00 24.38  0.00  0.00  0.00  9.04  0.00  0.00  0.00 29.87  0.00  0.00
E  5.42  6.44  0.00  3.10  7.00 12.91  0.00  0.00  0.00  4.40  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  5.35
G  0.00 22.81  0.00  4.66  7.88 47.91 55.51  0.00  5.35  0.00 92.78  0.00  0.00  0.00  0.00  0.00  0.00 80.31  0.00
H  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  4.59  0.00  0.00  0.00  0.00  0.00  0.00  9.08
I  0.00  0.00  6.16 45.33  0.00  0.00  0.00  4.63  2.                0.00  0.00 24.09 13.45 74.56  0.00  0.00 12.05
L  0.00  0.00 15.24 25.47  4.63  0.00  0.00 89.97  0.                9.04  9.08 13.96  0.00 10.10  0.00  0.00  0.00
K  6.95  2.57  4.16  0.00  0.00  0.00  0.00  0.00  5.               22.99  9.23  0.00  7.17  0.00 19.68  0.00  0.00
M  0.00  0.00  6.22  0.00  0.00  0.00  0.00  5.35  0.00  0.00  0.00 18.89  7.17  0.00  0.00  0.00  4.59  0.00  6.81
F  0.00  4.66  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  1.22  0.00  0.00  4.14  0.00  4.64
P 42.77  0.00 11.64  0.00 36.62  0.00  0.00  0.00  9.35 28.38  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00
S  2.94  4.69  4.46  0.00  7.17  0.00 29.02  0.00 19.05  0.00  0.00 27.88 16.52  0.00  0.00  0.00  0.00  0.00  0.00
T 16.88  6.20 29.34  0.00  5.03  0.00  0.00  0.00  7.17  9.41  0.00  4.05 23.01  0.00 36.20  0.00 13.75  0.00 29.77
W  0.00  0.00  6.43  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  6.35  0.00  0.00  0.00  0.00  0.00  0.00
Y  4.69 38.12  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  7.17  0.00  0.00 16.52  0.00  9.82  0.00  0.00
V 15.63  0.01  0.01  0.01  0.01 20.62  0.01  0.01  3.68 37.21  0.01  0.01  0.01 60.70 26.63 15.30  0.01  0.01 13.05
```

```
     P     Y     T     E     A     A     S     L     S     T     G     S     T     V     T     I     K     G     R
A  0.00  5.42 10.89 14.14 12.85  8.40  0.00  0.00  4.13  0.00  0.00  0.00  6.12  0.00  0.00  0.00  0.00 19.64  0.00
R  0.00  4.62  5.42  0.00  0.00  0.00  6.35  0.00  8.66  0.00  0.00  5.35  0.00  0.00  0.00  0.00  9.08  0.00 12.05
N  0.00  0.00  0.00  0.00  5.35  2.41  0.00  0.00  2.09  0.00  0.00  0.00  0.00  0.00  0.00  0.00  9.04  0.00  7.17
D  0.00  0.00  0.00  0.00  0.00  7.72  0.00  0.00  4.05 18.26  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00
C  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  4.63  0.00  7.17  0.00 13.45  0.00  0.00  0.00  0.00  0.00  0.00
Q  4.69  4.45  0.00  7.26 13.44  0.00  9.08  0.00 24.38  0.00  0.00  0.00  9.04  0.00  0.00  0.00 29.87  0.00  0.00
E  5.42  6.44  0.00  3.10  7.00 12.91  0.00  0.00  0.00  4.40  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  5.35
G  0.00 22.81  0.00  4.66  7.88 47.91 55.51  0.00  5.35  0.00 92.78  0.00  0.00  0.00  0.00  0.00  0.00 80.31  0.00
H  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  4.59  0.00  0.00  0.00  0.00  0.00  0.00  9.08
I  0.00  0.00  6.16 45.33  0.00  0.00  0.00  4.63  2.                0.00  0.00 24.09 13.45 74.56  0.00  0.00 12.05
L  0.00  0.00 15.24 25.47  4.63  0.00  0.00 89.97  0.                9.04  9.08 13.96  0.00 10.10  0.00  0.00  0.00
K  6.95  2.57  4.16  0.00  0.00  0.00  0.00  0.00  5.               22.99  9.23  0.00  7.17  0.00 19.68  0.00  0.00
M  0.00  0.00  6.22  0.00  0.00  0.00  0.00  5.35  0.00  0.00  0.00 18.89  7.17  0.00  0.00  0.00  4.59  0.00  6.81
F  0.00  4.66  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  1.22  0.00  0.00  4.14  0.00  4.64
P 42.77  0.00 11.64  0.00 36.62  0.00  0.00  0.00  9.35 28.38  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00
S  2.94  4.69  4.46  0.00  7.17  0.00 29.02  0.00 19.05  0.00  0.00 27.88 16.52  0.00  0.00  0.00  0.00  0.00  0.00
T 16.88  6.20 29.34  0.00  5.03  0.00  0.00  0.00  7.17  9.41  0.00  4.05 23.01  0.00 36.20  0.00 13.75  0.00 29.77
W  0.00  0.00  6.43  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  6.35  0.00  0.00  0.00  0.00  0.00  0.00
Y  4.69 38.12  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  0.00  7.17  0.00  0.00 16.52  0.00  9.82  0.00  0.00
V 15.63  0.01  0.01  0.01  0.01 20.62  0.01  0.01  3.68 37.21  0.01  0.01  0.01 60.70 26.63 15.30  0.01  0.01 13.05
```

```
H  0.40  0.46  0.36  0.36  0.31  0.28  0.22  0.21  0.                0.03  0.03  0.03  0.03  0.04  0.06  0.13  0.19
E  0.27  0.32  0.29  0.31  0.31  0.33  0.38  0.30  0.                0.20  0.84  0.89  0.90  0.86  0.73  0.46  0.31
C  0.33  0.22  0.35  0.33  0.38  0.39  0.40  0.49  0.77  0.86  0.84  0.77  0.13  0.08  0.07  0.10  0.21  0.41  0.50
```

Figure 5.2: Conceptual view of training SVM binary classifier for SS prediction (*h* = 5). (a) Training with sequence profile alone. (b) Training with sequence profile and SSCP.

## 5.4 Evaluation of Secondary Structure Prediction

The following metrics are used to measure the quality of SS prediction:

1. *The Three-state Single Residue Accuracy measure* (*$Q_3$*) has four components denoted by $Q_H$, $Q_E$, $Q_C$, and $Q_3$. For $s \in \{H, E, C\}$, $Q_s$ is the percentage of correctly predicted residues over all residues with observed label *s*. $Q_3$ is the overall accuracy calculated as the percentage of correctly predicted residues over the total in all three SS states.
2. *The Matthew's Correlation Coefficients* (*MCC*) [54] has three components denoted by $C_H$, $C_E$ and $C_C$. Each of them is calculated from a formula that accounts for both over- and under-predictions. A perfect prediction yields a value of 1, while a random prediction yields a near zero or even negative value.
3. *The Segment Overlap* (*SOV*) is designed to evaluate SS prediction on a non-per-residue basis. The original version, invented in 1994 by Rost et al. [55], has two serious problems. First, it yields un-normalized values that have no defined upper-bound, making it difficult for comparison. Second, the extension factor $\delta$ is miscalculated, resulting in inflated values that do not truly reflect the prediction quality. Fortunately, both problems have been corrected in a re-definition of SOV in 1999 [56] by Zemla et al. Unless specified otherwise, the corrected version is intended whenever SOV is mentioned in the remainder of this thesis.

## 5.5 Experiments and Results

### 5.5.1 Rotation Test

The data sets and method for the rotation test were as described in Section 4.7.1, except that the training set was also used for SSCP generation and SVM training in addition to motif cluster creation. Assignment rank *R* was set to 6. Parameters for SVM binary classifiers were 1.5 for error trade-off and 0.1 for $\gamma$ in the radial basis function used as the kernel [42]. SVM$^{light}$ [57] was extensively used throughout the experiment. The results are listed in Table 5.1.

Table 5.1: Prediction accuracy of SVM MAX trained without SSCP (top values) and trained with SSCP (bottom values) in the rotation test. Bolded pairs (3 instances) indicate a drop in accuracy after SSCP was used. A positive delta on the last row indicates an average improvement with SSCP (delta = average bottom value – average top value).

| **Training Set** | **Testing Set** | **$Q_3$ (%)** | **$Q_H$ (%)** | **$Q_E$ (%)** | **$Q_C$ (%)** | **$C_H$** | **$C_E$** | **$C_C$** | **SOV (%)** |
|---|---|---|---|---|---|---|---|---|---|
| BB55 | RS126 | 70.28<br>73.01 | 75.56<br>79.61 | 45.54<br>53.80 | **78.59**<br>**77.44** | 0.59<br>0.65 | 0.48<br>0.55 | 0.51<br>0.53 | 63.07<br>67.82 |
| | PP187 | 70.68<br>72.78 | 76.93<br>79.80 | 43.69<br>50.31 | **79.79**<br>**78.76** | 0.60<br>0.64 | 0.47<br>0.52 | 0.53<br>0.55 | 67.13<br>68.86 |
| | CB396 | 71.08<br>73.05 | 78.85<br>81.35 | 44.42<br>50.13 | **78.62**<br>**78.03** | 0.60<br>0.65 | 0.49<br>0.53 | 0.54<br>0.55 | 68.51<br>69.37 |
| RS126 | BB55 | 70.47<br>72.26 | 71.85<br>72.28 | 52.70<br>58.90 | 77.84<br>78.76 | 0.58<br>0.63 | 0.48<br>0.53 | 0.53<br>0.53 | 67.53<br>69.47 |
| | PP187 | 71.45<br>73.57 | 74.13<br>75.84 | 51.71<br>57.26 | 79.77<br>80.40 | 0.62<br>0.67 | 0.49<br>0.54 | 0.54<br>0.55 | 68.16<br>69.31 |
| | CB396 | 70.87<br>73.55 | 75.13<br>77.52 | 50.97<br>57.02 | 77.91<br>78.99 | 0.60<br>0.67 | 0.49<br>0.54 | 0.53<br>0.54 | 68.35<br>70.45 |
| PP187 | BB55 | 73.41<br>75.37 | 76.26<br>77.62 | 57.48<br>61.85 | 78.49<br>79.85 | 0.63<br>0.67 | 0.54<br>0.58 | 0.56<br>0.58 | 70.64<br>72.05 |
| | RS126 | 73.65<br>75.78 | 78.05<br>80.39 | 56.89<br>61.91 | 78.61<br>79.10 | 0.66<br>0.71 | 0.55<br>0.58 | 0.55<br>0.56 | 68.09<br>71.14 |
| | CB396 | 73.91<br>76.24 | 78.98<br>81.36 | 56.30<br>61.12 | 78.94<br>79.86 | 0.65<br>0.71 | 0.55<br>0.60 | 0.56<br>0.58 | 71.19<br>72.62 |
| CB396 | BB55 | 75.73<br>77.04 | 78.47<br>78.63 | 59.55<br>64.11 | 81.03<br>81.85 | 0.67<br>0.70 | 0.59<br>0.61 | 0.59<br>0.60 | 73.30<br>74.03 |
| | RS126 | 72.96<br>75.14 | 76.47<br>78.82 | 56.51<br>62.59 | 78.50<br>78.56 | 0.65<br>0.69 | 0.54<br>0.58 | 0.53<br>0.56 | 67.15<br>70.44 |
| | PP187 | 75.47<br>77.28 | 78.58<br>79.55 | 58.42<br>64.19 | 81.96<br>82.37 | 0.69<br>0.72 | 0.58<br>0.61 | 0.58<br>0.60 | 71.39<br>72.67 |
| Average | | 72.50<br>74.59 | 76.61<br>78.56 | 52.85<br>58.60 | 79.17<br>79.50 | 0.63<br>0.68 | 0.52<br>0.56 | 0.55<br>0.56 | 68.71<br>70.69 |
| Delta | | 2.09 | 1.95 | 5.75 | 0.33 | 0.05 | 0.04 | 0.01 | 1.98 |

### 5.5.2 Jackknife Test

As described in Section 4.7.2, data sets in the rotation test were likely to contain overlaps and yield unjust results, so a jackknife test was needed to evaluate the genuine contribution of SSCP. The data set (i.e. CB396) and method were as described in Section 4.7.2, except once again that each training set was used to generate SSCP and train SVM classifiers in addition to creating motif clusters. Assignment rank *R* and all parameters for SVM classifiers remained the same. The results from the jackknife test and their averages are shown in Table 5.2.

Table 5.2: Prediction accuracy of SVM MAX trained without SSCP (top values) and trained with SSCP (bottom values) in a ten-iteration jackknife test. Bolded pairs (3 instances) indicate a drop in accuracy after SSCP was used. A positive delta on the last row indicates an average improvement with SSCP (delta = average bottom value – average top value).

| **Iteration** | **$Q_3$ (%)** | **$Q_H$ (%)** | **$Q_E$ (%)** | **$Q_C$ (%)** | **$C_H$** | **$C_E$** | **$C_C$** | **SOV (%)** |
|---|---|---|---|---|---|---|---|---|
| 1 | 72.15<br>74.10 | 74.30<br>76.34 | 55.44<br>60.90 | 78.49<br>78.60 | 0.63<br>0.68 | 0.51<br>0.55 | 0.53<br>0.55 | 68.57<br>70.63 |
| 2 | 74.44<br>75.97 | **79.03**<br>**78.85** | 57.33<br>63.39 | 79.07<br>79.86 | 0.66<br>0.69 | 0.56<br>0.59 | 0.57<br>0.58 | 69.77<br>71.58 |
| 3 | 71.94<br>74.35 | 80.36<br>81.57 | 50.26<br>55.77 | 76.87<br>78.58 | 0.64<br>0.69 | 0.51<br>0.56 | 0.53<br>0.55 | 69.50<br>70.15 |
| 4 | 71.61<br>73.29 | 76.37<br>76.92 | 52.78<br>58.07 | 77.18<br>77.99 | 0.62<br>0.65 | 0.51<br>0.54 | 0.53<br>0.54 | 68.37<br>70.52 |
| 5 | 72.36<br>74.23 | 77.21<br>79.13 | 52.96<br>58.54 | **78.32**<br>**78.10** | 0.65<br>0.68 | 0.51<br>0.55 | 0.54<br>0.55 | 70.23<br>71.43 |
| 6 | 73.57<br>75.84 | 79.72<br>81.89 | 56.68<br>61.96 | 77.05<br>77.79 | 0.65<br>0.71 | 0.55<br>0.59 | 0.55<br>0.57 | 71.58<br>73.01 |
| 7 | 72.09<br>73.56 | 78.23<br>79.25 | 52.07<br>55.06 | 77.62<br>78.65 | 0.64<br>0.67 | 0.51<br>0.53 | 0.53<br>0.55 | 70.11<br>71.61 |
| 8 | 70.81<br>72.86 | 76.23<br>77.50 | 51.23<br>56.07 | 76.56<br>77.80 | 0.62<br>0.66 | 0.50<br>0.53 | 0.51<br>0.53 | 66.54<br>68.57 |
| 9 | 73.13<br>75.01 | 76.25<br>77.82 | 56.36<br>62.51 | **79.52**<br>**79.33** | 0.64<br>0.68 | 0.55<br>0.59 | 0.55<br>0.56 | 70.78<br>72.92 |
| 10 | 70.70<br>72.86 | 74.93<br>77.24 | 50.48<br>56.39 | 77.72<br>77.72 | 0.61<br>0.65 | 0.49<br>0.53 | 0.52<br>0.54 | 66.34<br>67.97 |
| Average | 72.28<br>74.21 | 77.26<br>78.65 | 53.56<br>58.87 | 77.84<br>78.44 | 0.64<br>0.68 | 0.52<br>0.56 | 0.53<br>0.55 | 69.18<br>70.84 |
| Delta | 1.93 | 1.39 | 5.31 | 0.60 | 0.04 | 0.04 | 0.02 | 1.66 |

The improvements (i.e. the deltas) in Table 5.1 are generally larger than those in Table 5.2. This is within expectation as overlaps between training and testing sets in the rotation test helped generate more reliable SSCP, which in turn contributed to greater improvements in SS prediction. The differences, however, are not significant. For instance, there has been a drop of only 8% in $Q_3$ and 16% in SOV going from the rotation test to the jackknife test. Hence, while being slightly biased, results from the rotation test can be considered valid.

### 5.5.3 Discussion

Both tests have shown that by combining SSCP with sequence profile for training and classification, SVM MAX predictor showed improvements in all $Q_3$, MCC and SOV measures. Specifically, SSCP contributed to an average $Q_3$ improvement of 2.09% (from 72.50% to 74.59%) in the rotation test and 1.93% (from 72.28% to 74.21%) in the jackknife test. It did so by boosting the prediction accuracy for helixes and strands, the latter in particular. In other words, SSCP helped the predictor be more certain when determining if a residue was part of a helix or strand. Moreover, the use of SSCP also resulted in visible improvements in all aspects of MCC and SOV, regardless of tests and data sets.

Unfortunately, improvements to $Q_C$ and $C_C$ were only minimal. After all, clusters could only capture regions with strong sequence-structure correlations, a condition excluding most coils. Consequently, cluster assignments made to segments along coil regions were mostly incorrect, leading to unreliable SS confidence values and subsequently the negligible increase in coil prediction accuracies.

# Chapter 6

# Conclusion and Future Work

## 6.1 Approximation Algorithm for Tertiary Structure Prediction

Recall from Section 4.6 that the DP algorithm for local tertiary structure prediction has a runtime of O($n$ $L^3$ $(R+1)^{L+1}$), where $n$ is the length of the target protein, $L$ is the segment length, and $R$ is the assignment rank. The exponential term restricts $R$ to a small value such as 3 in this study. Note that a larger $R$ means a larger conformational search space (see Figure 4.4(a)) and possibly better predictions as a result. Unfortunately, while a large value of $R$ such as 10 or more might be desirable, it would lead to a prohibitive execution time.

To draw a balance, a viable option would be to develop an approximate DP algorithm that sacrifices optimality for an execution time allowing larger values of $R$. An example that has been considered is a "greedy" DP algorithm. For each assignment $\alpha$ to be appended, the algorithm keeps track of the $R$ * $L$ assignment sets such that the last assignment in every set touches $\alpha$. The greedy nature comes in when $\alpha$ is appended to the set such that the resultant set yields the highest objective score. While the optimality for the final prediction is lost, the runtime requirement is only O($nR^2L^3$). Unfortunately, the option has not been further investigated since proving an approximation guarantee for the algorithm would be another research topic on its own.

## 6.2 A Better DP Objective Function

There is one aspect regarding the current DP objective function (i.e. Equation (4.3)) that might require some major refinement. For convenience, the objective function is restated in Equation (6.1) below. Please refer to Section 4.5 for further details.

$$F(X) = q\sum_{i=0}^{n-1} score(X, i) - \sum_{i=0}^{n-1} conflict(X, i) \tag{6.1}$$

Function *conflict* might not always have appropriately reflected the structural disagreement between overlapping assignments in some circumstance. Recall from Section 4.5 that *conflict*(*X*, *i*) returns the average dihedral angle difference between all pairs of overlapping assignments in *X* covering position *i*. Assume for now that there are only two assignments $\alpha_1$ and $\alpha_2$ in *X* covering position *i*, and at that position the *phi* angles are $0^o$ and $100^o$ for $\alpha_1$ and $\alpha_2$ respectively[4]. By definition, *conflict*(*X*, *i*) returns $|100^o – 0^o| / 1 = 100^o$. If another assignment $\alpha_3$ is subsequently appended to *X* to produce *X'* such that it covers position *i* with a *phi* angle of $50^o$, then *conflict*(*X'*, *i*) only returns $(|100^o – 0^o| + |100^o – 50^o| + |50^o – 0^o|) / 3 = 66.67^o$. In other words, the addition of $\alpha_3$ has "harmonized" $\alpha_1$ and $\alpha_2$ by partially hiding their serious structural disagreement, which is certainly flawed. Improving the objective function by minimizing or even eliminating the deficiency is the key to achieving better predictions.

[4] WLOG, *psi* angles have been ignored for simplicity.

## 6.3 Prediction using PSI-BLAST Profiles

Aside from HSSP-derived sequence profiles, this study has also been conducted using PSI-BLAST profiles, but only briefly because of a restriction imposed by the clustering algorithm described in Chapter 3. The distance function shown in Equation (3.1) assumes that all profile entries are non-negative and all entries for every residue sum to 1. Unfortunately, PSI-BLAST profiles contain log-odds entries that violate all these assumptions. Although PSI-BLAST does provide a frequency profile in its output as depicted in Figure 6.1, using the frequency profile for clustering and prediction have only produced results similar to the ones obtained with HSSP-derived profiles. A possible reason for the disappointment is that the real strength of PSI-BLAST lies in its sophisticated mechanism behind generating unbiased log-odds profiles. Consequently, PSI-BLAST will not contribute to any significant improvement unless the clustering algorithm can be made to take advantage of its log-odds profiles. Despite a promising direction for enhancement, it is not pursued at present as it requires making substantial changes.

```
   A  R  N  D  C  Q  E  G  H  I  L  K  M  F  P  S  T  W  Y  V :  A  R  N  D  C  Q  E  G  H  I  L  K  M  F  P  S  T  W  Y  V
M -3 -3 -4 -5 -3 -3 -4 -5 -4  0  4 -3  8 -1 -4 -4 -3 -3 -3 -1 :  0  0  0  0  0  0  0  0  0  0 43  0 57  0  0  0  0  0  0  0
K  0  3 -1 -3 -5  3 -1 -3  0 -1 -4  6 -3 -5 -3 -2 -1 -5 -4 -3 :  9 12  2  0  0 16  0  0  1  4  0 54  0  0  0  0  2  0  0  0
L -2  0 -5 -5 -3  0 -4 -5 -4  2  5 -4  0 -2 -5 -2 -3 -4 -3 -1 :  3  5  0  0  0  6  0  0  0 14 68  0  0  0  0  4  0  0  0  0
F -3  1  0 -2 -4  1 -3  1 -2 -1 -2  0 -1  2  4  1  0 -4 -1 -3 :  0 10  5  2  0  7  0 10  1  4  4  7  2 11 22 10  5  0  2  0
A  2 -3 -2 -2  5 -3 -1 -4 -2  0  2 -3 -2 -3 -1  0 -2 -4 -2  1 :23  1  2  2 13  0  4  0  1  6 26  0  0  0  3  5  1  0  1 11
Q  0  1  1 -1  0  1  0  4  3 -5 -4  0 -4 -4 -1 -2  0 -4  0 -4 :  9  8  8  2  2  6  5 33  7  0  1  4  0  0  3  0  6  0  3  0
G -3  1  2  1  2 -1  0  4 -3 -2 -2  2 -4 -5 -4 -1 -2 -5 -5 -4 :  0  7 11  7  5  2  5 40  0  2  5 12  0  0  0  3  1  0  0  0
T -3  6 -2 -2 -5 -1 -3  0  1 -3 -3  3  2  0 -4 -3 -2  0  0  1 :  1 44  1  2  0  2  0  6  3  0  1 16  5  4  0  0  2  1  3 10
S -2 -2 -3 -3  0 -1  1 -3 -2  0  3  2 -1  1  1  0  1 -4 -3  1 :  1  1  1  1  2  1  9  1  1  4 27 16  1  6  6  6  8  0  0  9
```

Figure 6.1: PSI-BLAST profiles in a PSP output file, where the dotted line separates log-odds profile (left) from frequency profile (right)

## 6.4 Motifs Capturing Long-Range Residue Interactions

The current sequence-structure motifs can only capture local inter-residue interactions, so they are not very helpful for beta-sheet prediction. In the long run, the solution is to study non-local motifs formed primarily by interactions between distant residues. Conceptually, a non-local motif of size *n* would comprise *n* local sequence-structure motifs and up to (*n*)(*n*–1)/2 interactions among the *n* motifs. Figure 6.2 shows an instance of non-local motif *x* of size *n* = 2.

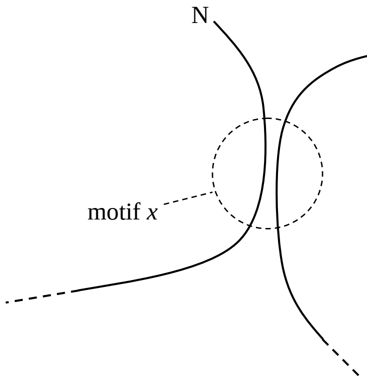


Figure 6.2: Motif x capturing the distant interaction between two stretches of the same protein, where ‘N’ and ‘C’ denote the N and C termini respectively

One method for discovering non-local motifs of size *n* is to extract all local sequence-structure motifs, select all *n*-tuples of mutually interacting local motifs, and perform clustering on the resultant *n*-tuples. The primary issue with the extraction of non-local motifs is that there might not be sufficient training data (i.e. resolved protein structures) to give rise to any significant motifs, even for *n* = 2. Other issues may also arise such as those concerning the measurement of distances between *n*-tuples of segments and the determination of a suitable similarity threshold. In spite of all the issues, extraction of non-local motifs is worth exploring as a systemic way for categorizing and analyzing long-range interactions. In the future, non-local motifs might even be combined with the local ones to directly predict global tertiary structures.

## 6.5 Conclusion

The partition of short protein segments into clusters of local sequence-structure motifs has profound applications. It effectively reveals the composition and fold characterizing each motif, enabling the inference of structural formation and functional role. Besides biological studies, these motif clusters achieve discretization of protein conformational space and provide an adequate mapping between sequence and structure, all contributing to the success of their employment to both secondary and tertiary structure prediction. The promising results obtained in this study could mark the beginning of a wide range of potential applications for motif clusters, which include fold recognition, domain detection, functional annotation, and structural correction for NMR and X-ray Crystallography.

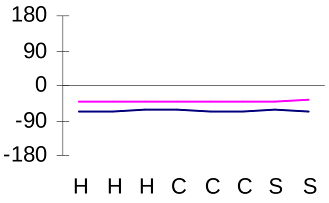
# Appendix A

# Listing of Sequence-Structure Motifs

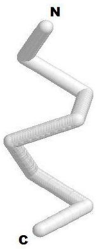
This appendix presents some significant sequence-structure motifs discovered by clustering the set of proteins known as CB396 [49]. Each entry shows the number of segments exhibiting the motif, the dihedral angle plot, the log-odds profile, and the 3D backbone drawing. In each dihedral angle plot, the *phi* angle is denoted by a blue (dark) line and the *psi* angle is denoted by a magenta (light) line. Please refer to Section 3.6 for further details.

## α-Helices

511

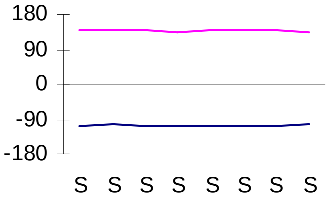


| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| A | • | 1 | • | • | 1 | 1 | 1 | • |
| R | • | • | • | 1 | • | -1 | • | 1 |
| N | • | -1 | -2 | • | • | -1 | -1 | • |
| D | 1 | • | -1 | • | • | -2 | -1 | 1 |
| C | -2 | -1 | • | -1 | -1 | • | • | -2 |
| Q | 1 | • | • | 1 | 1 | • | • | 1 |
| E | 1 | 1 | -1 | 1 | 1 | -1 | • | 1 |
| G | -1 | -1 | -2 | -1 | -1 | -2 | -2 | -1 |
| H | • | • | -1 | • | • | • | • | • |
| I | -1 | • | 1 | -1 | -1 | 1 | • | -1 |
| L | -1 | • | 1 | • | -1 | 1 | 1 | -1 |
| K | 1 | • | -1 | 1 | 1 | -1 | • | 1 |
| M | -1 | • | 1 | • | -1 | 1 | 1 | -1 |
| F | -2 | • | • | -1 | -1 | • | • | -2 |
| P | • | -1 | -2 | -1 | -1 | -2 | -2 | -1 |
| S | • | • | -1 | • | • | -1 | • | • |
| T | • | • | -1 | -1 | • | -1 | -1 | • |
| W | -1 | • | • | -1 | -1 | • | • | -1 |
| Y | -1 | • | • | -1 | -1 | • | • | -1 |
| V | -1 | • | 1 | -1 | -1 | • | • | -1 |

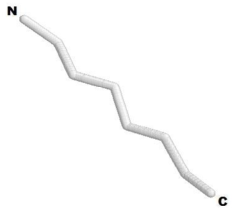


## β-Strands

196

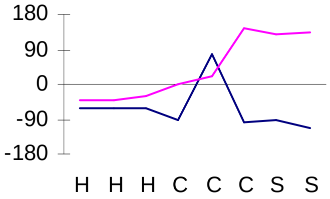


| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| A | • | • | • | • | • | -1 | • | • |
| R | • | -1 | -1 | • | • | • | -1 | • |
| N | -1 | • | • | • | -1 | 2 | • | 1 |
| D | • | • | -2 | • | -1 | • | • | • |
| C | -1 | -1 | -3 | -2 | -2 | -2 | -2 | -1 |
| Q | • | • | -1 | • | -1 | -1 | 1 | • |
| E | -1 | 1 | -1 | • | • | -1 | 1 | • |
| G | -2 | -1 | -1 | -1 | • | 2 | • | • |
| H | -1 | -1 | -1 | 1 | • | • | 1 | -2 |
| I | 2 | • | 1 | • | -2 | -3 | -2 | -1 |
| L | • | • | 1 | • | • | -1 | -2 | -1 |
| K | • | • | -1 | • | • | • | -1 | • |
| M | • | • | • | 1 | 1 | -1 | • | -2 |
| F | -1 | -1 | 1 | • | • | -1 | • | -1 |
| P | -1 | • | -1 | 1 | 2 | • | • | 1 |
| S | -1 | • | • | • | • | • | 1 | 1 |
| T | • | 1 | -1 | -1 | • | -1 | • | 1 |
| W | • | • | -2 | -2 | 1 | -3 | -2 | -1 |
| Y | • | -1 | -2 | -1 | • | -2 | • | • |
| V | 1 | • | 1 | • | -1 | -2 | -1 | • |

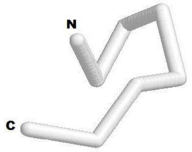


## Helix C-Caps

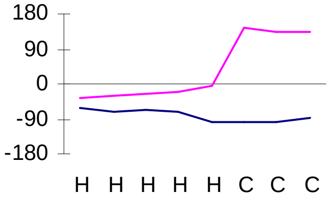
58



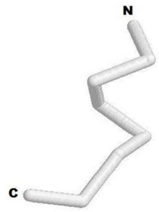
| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| A | • | 1 | • | • | 1 | 1 | 1 | • |
| R | • | • | • | 1 | • | -1 | • | 1 |
| N | • | -1 | -2 | • | • | -1 | -1 | • |
| D | 1 | • | -1 | • | • | -2 | -1 | 1 |
| C | -2 | -1 | • | -1 | -1 | • | • | -2 |
| Q | 1 | • | • | 1 | 1 | • | • | 1 |
| E | 1 | 1 | -1 | 1 | 1 | -1 | • | 1 |
| G | -1 | -1 | -2 | -1 | -1 | -2 | -2 | -1 |
| H | • | • | -1 | • | • | • | • | • |
| I | -1 | • | 1 | -1 | -1 | 1 | • | -1 |
| L | -1 | • | 1 | • | -1 | 1 | 1 | -1 |
| K | 1 | • | -1 | 1 | 1 | -1 | • | 1 |
| M | -1 | • | 1 | • | -1 | 1 | 1 | -1 |
| F | -2 | • | • | -1 | -1 | • | • | -2 |
| P | • | -1 | -2 | -1 | -1 | -2 | -2 | -1 |
| S | • | • | -1 | • | • | -1 | • | • |
| T | • | • | -1 | -1 | • | -1 | -1 | • |
| W | -1 | • | • | -1 | -1 | • | • | -1 |
| Y | -1 | • | • | -1 | -1 | • | • | -1 |
| V | -1 | • | 1 | -1 | -1 | • | • | -1 |



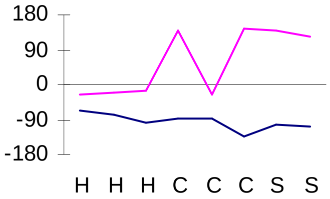
41



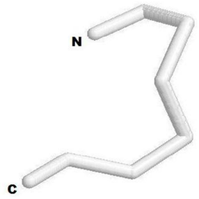
| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| A | • | • | • | • | • | • | • | • |
| R | 1 | • | 1 | • | • | • | • | • |
| N | • | -1 | • | • | 1 | • | • | • |
| D | 1 | -1 | • | 1 | • | -1 | • | 1 |
| C | -3 | -1 | -1 | -3 | -1 | -1 | -1 | • |
| Q | 1 | • | • | 1 | • | • | 1 | -1 |
| E | 1 | • | • | 1 | • | • | • | • |
| G | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 |
| H | -1 | • | • | • | 1 | • | • | • |
| I | -1 | 1 | • | -1 | -1 | 1 | -1 | • |
| L | • | 1 | 1 | -1 | • | 1 | -1 | • |
| K | 1 | • | • | 1 | 1 | • | 1 | • |
| M | • | • | -1 | • | • | • | -1 | • |
| F | • | • | • | • | • | • | -1 | • |
| P | • | -1 | • | • | • | -1 | 2 | 1 |
| S | • | • | • | • | • | • | • | • |
| T | -1 | -1 | • | -1 | • | • | • | • |
| W | -1 | 1 | -1 | • | -2 | -3 | -1 | 1 |
| Y | -1 | • | • | • | • | • | • | -1 |
| V | -1 | • | • | -2 | -1 | 1 | -1 | • |



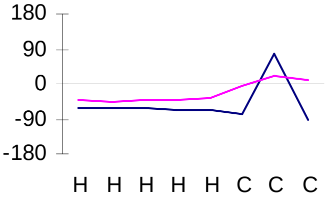
22



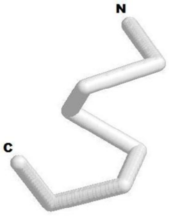
| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| A | 1 | • | -1 | 1 | • | • | -1 | -1 |
| R | • | • | 1 | -2 | • | • | -1 | • |
| N | -1 | • | 1 | -1 | • | • | -1 | -2 |
| D | -1 | • | • | -1 | 1 | • | -1 | -2 |
| C | -1 | -3 | • | • | -2 | -1 | -1 | • |
| Q | • | 1 | 1 | -2 | • | • | • | • |
| E | • | 1 | • | -1 | • | 1 | • | -1 |
| G | -2 | -1 | -1 | -1 | • | -1 | -2 | -1 |
| H | -1 | • | 1 | • | • | 1 | • | • |
| I | • | -1 | -1 | 1 | -2 | • | 1 | 1 |
| L | 1 | • | • | 1 | -1 | -1 | • | 1 |
| K | • | 1 | 1 | -1 | 1 | 1 | • | -1 |
| M | 1 | -1 | • | -1 | -2 | -1 | -1 | -1 |
| F | • | • | • | • | -2 | • | • | • |
| P | • | • | -1 | -1 | 1 | • | 2 | 1 |
| S | • | 1 | -1 | • | • | • | • | • |
| T | • | -1 | -1 | -1 | • | -1 | • | -1 |
| W | -1 | -2 | • | -1 | -1 | -1 | -2 | • |
| Y | • | • | 1 | • | -2 | 1 | • | • |
| V | 1 | -1 | -1 | 1 | -1 | • | 1 | 1 |



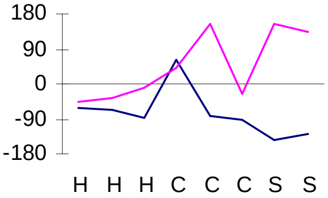
21



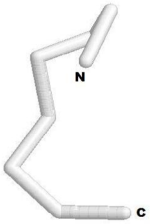
|   | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| A | • | • | 1 | -1 | 1 | 1 | -1 | • |
| R | • | • | 1 | -1 | • | • | • | -1 |
| N | • | 1 | -1 | • | 1 | 1 | 1 | 1 |
| D | -1 | 1 | • | -1 | • | 1 | • | 1 |
| C | -1 | -3 | -1 | • | -5 | -4 | -2 | -5 |
| Q | • | 1 | 1 | -1 | 1 | 1 | • | 1 |
| E | • | 1 | • | • | 1 | 1 | • | 1 |
| G | -1 | -1 | • | -1 | -2 | -1 | 2 | • |
| H | • | • | -1 | 1 | -1 | -1 | • | -1 |
| I | 1 | -3 | -1 | 1 | • | -1 | -2 | -2 |
| L | • | -1 | 1 | • | • | -1 | -1 | -2 |
| K | • | 1 | • | • | 1 | • | • | 1 |
| M | 1 | • | 1 | 1 | • | • | -2 | -1 |
| F | • | • | • | 1 | -1 | -3 | -2 | -1 |
| P | • | -2 | -1 | -2 | -2 | • | -1 | 1 |
| S | -1 | • | • | -1 | -1 | • | • | • |
| T | -1 | -1 | • | -1 | -1 | • | -1 | • |
| W | -1 | -1 | -2 | 2 | -3 | -2 | -5 | -2 |
| Y | • | -1 | • | 2 | • | • | -1 | -1 |
| V | 1 | -2 | -1 | • | -1 | -2 | -1 | -1 |



16

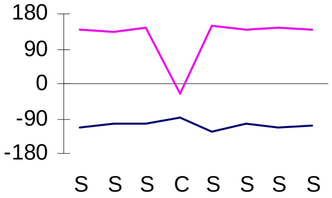


|   | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| A | • | • | • | -1 | 1 | • | 1 | • |
| R | 1 | 1 | • | • | • | -1 | -1 | -3 |
| N | • | • | • | 1 | -1 | • | -1 | -11 |
| D | • | • | -2 | 1 | -2 | 2 | • | -11 |
| C | -2 | -2 | -1 | -3 | 1 | -4 | -1 | -1 |
| Q | 1 | 1 | • | 1 | -3 | 1 | -1 | -7 |
| E | 1 | 1 | -2 | • | -2 | • | • | -5 |
| G | -1 | -1 | -2 | 2 | -2 | -2 | 1 | -4 |
| H | • | • | 2 | • | • | • | -1 | -3 |
| I | -1 | -1 | -1 | -3 | • | -1 | -1 | 2 |
| L | • | -1 | • | -2 | • | • | • | • |
| K | 2 | 1 | • | 1 | • | 1 | -1 | -11 |
| M | -1 | • | • | -2 | -2 | -3 | • | • |
| F | -3 | -2 | 1 | -4 | • | -2 | -1 | • |
| P | -2 | -2 | -1 | • | 2 | -1 | -3 | -4 |
| S | • | • | -1 | -1 | -1 | • | • | -1 |
| T | -1 | • | 1 | -2 | -2 | • | -1 | -1 |
| W | -4 | -3 | -1 | -11 | -2 | -1 | -3 | -3 |
| Y | -1 | 1 | 1 | -2 | -2 | -1 | -1 | -2 |
| V | -1 | -1 | • | -2 | 1 | -1 | 1 | 3 |

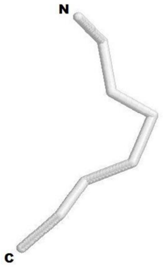


## Inter-Strand Turns and Hairpins

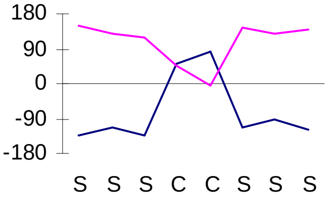
26



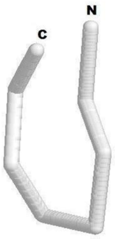
|   | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| A | 1 | • | • | • | • | • | • | • |
| R | -1 | • | -1 | -1 | • | • | • | -1 |
| N | -1 | • | • | • | 1 | -1 | -2 | • |
| D | -1 | • | -1 | 1 | • | -1 | -1 | -1 |
| C | -2 | -1 | -1 | -2 | -5 | -4 | -2 | -1 |
| Q | • | 1 | -1 | • | 1 | • | • | • |
| E | • | 1 | -1 | 1 | 1 | • | • | • |
| G | -1 | -2 | • | • | • | -2 | -1 | -2 |
| H | • | • | • | -1 | 1 | -1 | -1 | • |
| I | • | • | • | -1 | -1 | 1 | 1 | 1 |
| L | • | • | • | -1 | -2 | 1 | • | • |
| K | • | • | • | 1 | 1 | • | • | • |
| M | -1 | • | -1 | -1 | -2 | -1 | • | -1 |
| F | 1 | • | -1 | -2 | -1 | • | 1 | • |
| P | 1 | • | • | • | -1 | 1 | • | • |
| S | • | • | 1 | 1 | 1 | • | • | • |
| T | • | 1 | • | 1 | • | • | • | 1 |
| W | • | • | -4 | -3 | • | -2 | -4 | -2 |
| Y | • | • | 1 | -1 | • | -2 | 1 | • |
| V | • | • | 1 | • | -1 | 1 | 1 | 1 |



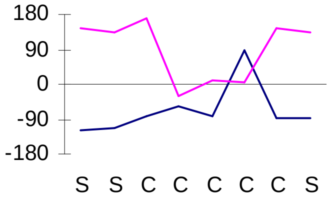
25



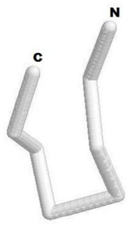
|   | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| A | • | • | -1 | -1 | -1 | • | -1 | -1 |
| R | • | • | • | • | • | 1 | -1 | 1 |
| N | -1 | • | • | 2 | 1 | 1 | -1 | -1 |
| D | -2 | 1 | 1 | 1 | 1 | • | -1 | -1 |
| C | • | -1 | -1 | -4 | -3 | -3 | -1 | 1 |
| Q | • | 1 | • | • | -1 | 1 | • | • |
| E | -1 | • | • | • | • | 1 | • | • |
| G | -2 | -1 | • | 1 | 2 | • | -1 | -2 |
| H | -1 | • | • | -1 | -1 | • | -2 | • |
| I | 1 | • | • | -3 | -2 | -2 | 1 | 1 |
| L | • | • | • | -3 | -2 | -2 | 1 | • |
| K | -1 | • | • | 1 | • | 2 | • | • |
| M | • | • | • | -2 | -3 | -1 | • | -1 |
| F | 1 | • | • | -2 | -2 | -1 | • | 1 |
| P | -1 | -2 | -1 | -1 | -1 | -1 | • | -2 |
| S | -1 | • | • | • | • | • | -1 | -1 |
| T | 1 | • | • | -1 | -1 | • | 1 | • |
| W | -2 | 1 | 2 | -4 | -2 | -2 | 1 | -1 |
| Y | 1 | • | 1 | -2 | -1 | • | • | 1 |
| V | 1 | 1 | • | -3 | -1 | • | 1 | 1 |



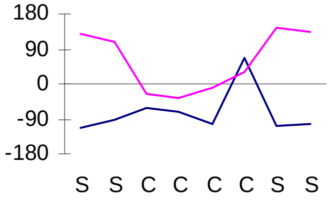
20



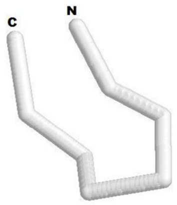
| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| A | • | -2 | -2 | • | -1 | -2 | • | -1 |
| R | -1 | • | -1 | • | • | -1 | 1 | • |
| N | -1 | -1 | 1 | 1 | 2 | 1 | 1 | -1 |
| D | -2 | -2 | 2 | 1 | 2 | -1 | • | -1 |
| C | • | -1 | -1 | -1 | -4 | -2 | -2 | • |
| Q | -2 | • | • | • | 1 | -1 | 1 | -1 |
| E | -1 | • | • | 1 | 1 | -1 | 1 | -1 |
| G | -1 | -1 | -1 | • | • | 3 | • | -2 |
| H | -1 | • | -1 | -1 | • | -1 | • | • |
| I | 1 | 1 | -2 | -2 | -2 | -2 | -1 | • |
| L | 1 | 1 | -1 | -2 | -2 | -1 | -1 | • |
| K | -1 | 1 | • | 1 | • | • | 1 | • |
| M | • | • | -1 | -2 | -3 | -1 | -1 | • |
| F | 1 | • | -1 | -1 | -3 | -2 | -2 | • |
| P | -2 | -1 | -1 | 1 | -2 | -1 | -1 | 2 |
| S | -1 | -1 | • | • | • | -1 | • | -1 |
| T | -1 | • | 2 | • | 1 | -1 | • | • |
| W | 1 | -2 | • | -11 | -11 | -2 | -3 | • |
| Y | 1 | • | -1 | -2 | -2 | -2 | -2 | • |
| V | 1 | 1 | -2 | -1 | -3 | -2 | -1 | 1 |



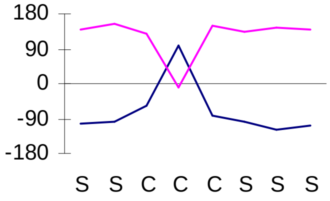
20



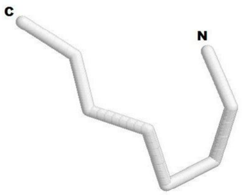
| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| A | -1 | -1 | • | • | -1 | -2 | -1 | -1 |
| R | • | -1 | • | • | • | 1 | 2 | • |
| N | -2 | 2 | • | 1 | 1 | 1 | -1 | -1 |
| D | -2 | 2 | • | 1 | 1 | 1 | -1 | -1 |
| C | -1 | -2 | -2 | -2 | -2 | -5 | -1 | -2 |
| Q | -1 | -1 | • | 1 | -1 | 1 | 1 | • |
| E | -1 | -1 | 1 | • | • | • | 1 | • |
| G | -2 | -1 | -1 | • | • | 2 | -2 | -2 |
| H | • | • | -1 | • | • | • | -1 | -1 |
| I | 1 | -2 | -1 | -1 | -3 | -3 | -1 | 1 |
| L | -1 | -2 | • | -1 | -2 | -2 | • | 1 |
| K | • | • | -1 | 1 | 1 | 1 | 2 | • |
| M | -1 | -2 | • | -2 | -1 | • | -3 | 1 |
| F | 1 | -1 | -1 | -2 | -2 | -1 | -2 | • |
| P | • | -1 | 2 | -1 | -1 | -2 | -2 | -3 |
| S | -1 | • | • | 1 | 1 | • | • | -1 |
| T | • | -1 | • | • | 2 | -2 | • | 1 |
| W | 1 | 1 | -1 | -1 | -1 | -2 | -2 | • |
| Y | 2 | -1 | • | -2 | -3 | -2 | -1 | • |
| V | 1 | -2 | • | -1 | -2 | -3 | -1 | 1 |



18

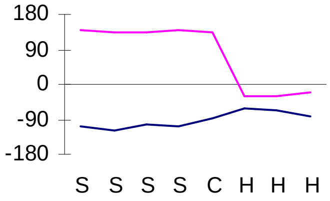


| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| A | • | • | • | -1 | • | -1 | • | • |
| R | -1 | • | -1 | -1 | -1 | 1 | -3 | -1 |
| N | -3 | • | • | 1 | • | -1 | -3 | -1 |
| D | -4 | -1 | • | -1 | 2 | -2 | -3 | • |
| C | -1 | -5 | -2 | -3 | -1 | -1 | -2 | -4 |
| Q | -2 | 1 | • | -3 | 2 | 1 | -2 | • |
| E | -2 | 1 | 1 | -1 | 1 | • | -2 | 1 |
| G | -1 | • | -2 | 3 | -1 | -2 | -3 | -1 |
| H | -2 | • | -1 | -1 | • | -1 | -2 | • |
| I | 1 | -2 | • | -5 | -2 | • | 2 | • |
| L | 1 | -2 | • | -3 | -3 | -1 | 1 | 1 |
| K | -1 | 2 | • | -1 | • | 1 | -2 | • |
| M | • | -2 | -3 | -1 | • | -1 | • | -1 |
| F | -1 | -1 | • | -4 | -3 | • | -1 | -2 |
| P | • | 1 | 2 | -2 | -3 | • | -4 | • |
| S | -1 | 1 | • | -2 | 1 | • | -2 | • |
| T | -1 | • | -1 | -3 | • | 1 | -2 | 1 |
| W | -3 | -4 | -11 | -3 | -1 | • | -3 | -2 |
| Y | -1 | -1 | -1 | -6 | • | • | -3 | -1 |
| V | 2 | -1 | • | -4 | -2 | • | 3 | • |

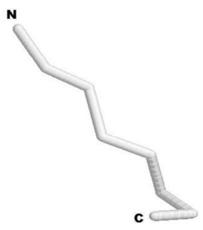


## Strand-Helix Turns

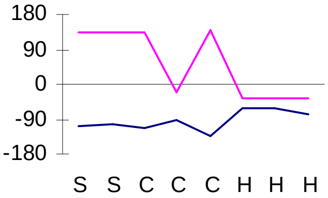
63



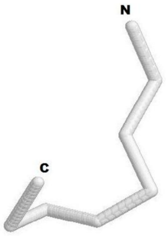
| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| A | • | -1 | • | • | -1 | • | • | • |
| R | • | -1 | • | • | -1 | • | • | • |
| N | • | -1 | • | -1 | 1 | • | • | • |
| D | • | -1 | • | -1 | 2 | 1 | 1 | 1 |
| C | -1 | • | • | • | -1 | -1 | -2 | -2 |
| Q | • | • | • | -1 | -1 | • | • | 1 |
| E | • | • | • | -1 | • | 1 | 1 | 1 |
| G | -1 | -1 | -2 | -1 | -1 | -1 | • | -1 |
| H | • | • | • | • | • | -1 | • | • |
| I | • | • | • | 1 | -2 | -1 | -2 | -1 |
| L | • | • | -1 | 1 | -1 | • | -1 | • |
| K | • | • | • | -1 | • | • | 1 | • |
| M | • | • | • | • | -1 | -1 | -1 | -1 |
| F | • | 1 | • | 1 | -1 | -1 | -1 | • |
| P | • | • | • | • | 1 | 2 | • | -1 |
| S | • | • | • | -1 | 1 | • | • | • |
| T | • | • | • | -1 | 1 | • | • | • |
| W | • | • | -1 | 1 | -3 | -1 | -2 | • |
| Y | • | 1 | • | 1 | -2 | -1 | -1 | • |
| V | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 |



29

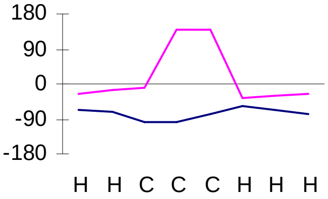


|   | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| A | • | 1 | • | • | -1 | • | 1 | • |
| R | 1 | • | -1 | • | -2 | • | • | • |
| N | -1 | -1 | -2 | • | 2 | -1 | • | -1 |
| D | -1 | -1 | -1 | • | 2 | • | 1 | 1 |
| C | -2 | -1 | • | -3 | -2 | -2 | -3 | -2 |
| Q | 1 | • | -1 | -1 | -1 | • | • | 1 |
| E | • | • | -1 | • | • | • | 2 | 2 |
| G | -2 | -1 | -1 | -1 | -1 | -1 | -1 | -2 |
| H | 1 | -1 | • | • | • | -1 | -1 | 1 |
| I | 1 | • | 1 | -1 | -2 | -1 | -1 | -1 |
| L | • | • | • | -1 | -3 | • | -1 | -1 |
| K | • | • | -1 | • | -1 | • | • | • |
| M | • | -1 | -1 | • | -2 | • | -1 | -1 |
| F | • | • | • | • | -3 | -1 | -2 | -1 |
| P | -1 | 1 | • | 1 | • | 2 | -1 | -2 |
| S | -1 | • | -1 | • | 1 | • | • | • |
| T | -1 | • | • | 1 | 1 | • | • | • |
| W | -1 | • | -1 | -1 | -1 | -2 | -1 | • |
| Y | • | • | 1 | -1 | -1 | -1 | -1 | -1 |
| V | 1 | 1 | 2 | • | -2 | -1 | -1 | -1 |

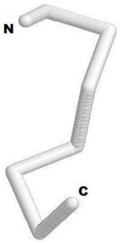


## Inter-Helix Turns

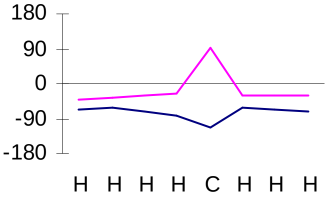
44



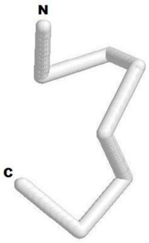
|   | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| A | • | 1 | • | • | -1 | • | • | • |
| R | -1 | 1 | • | -1 | -1 | • | -1 | • |
| N | -1 | • | • | • | 1 | • | • | • |
| D | -1 | 1 | • | • | 2 | • | 1 | 1 |
| C | • | -3 | • | • | -1 | -3 | -2 | • |
| Q | • | • | • | • | • | • | 1 | 1 |
| E | • | 1 | • | -1 | • | • | 1 | 1 |
| G | -1 | -1 | -1 | -1 | • | -1 | -1 | -1 |
| H | -2 | • | • | • | • | -1 | -1 | • |
| I | • | -1 | • | 1 | -2 | • | -2 | -2 |
| L | 1 | -1 | • | 1 | -1 | • | -1 | -1 |
| K | • | 1 | • | -1 | • | • | • | • |
| M | 1 | • | 1 | 1 | -1 | • | -2 | -1 |
| F | • | -1 | • | • | -2 | -1 | -2 | -1 |
| P | • | • | -2 | -2 | 2 | 1 | • | -1 |
| S | • | • | • | • | 1 | • | • | • |
| T | • | • | • | • | • | • | • | 1 |
| W | 1 | -2 | • | -1 | -2 | -1 | -2 | • |
| Y | • | -1 | • | • | -1 | • | -1 | • |
| V | • | -1 | • | 1 | -1 | -1 | -1 | • |



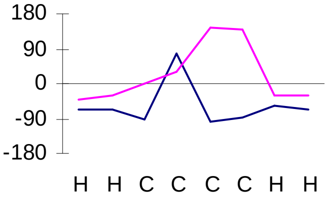
40



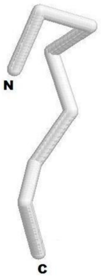
|   | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| A | • | • | 1 | 1 | -1 | • | • | • |
| R | • | • | 1 | • | -1 | -1 | • | • |
| N | -1 | -1 | • | • | 2 | -1 | • | • |
| D | • | -2 | • | • | 1 | • | 1 | • |
| C | -5 | -1 | • | -2 | • | -2 | -4 | -1 |
| Q | • | • | 1 | 1 | • | • | 1 | • |
| E | -1 | • | 1 | 1 | • | • | 1 | • |
| G | -2 | -2 | -2 | • | -2 | -1 | • | -2 |
| H | -1 | • | • | • | 1 | -1 | -1 | • |
| I | • | 1 | -1 | -2 | -1 | -2 | -1 | • |
| L | 1 | 1 | -1 | -1 | -1 | -1 | -1 | 1 |
| K | • | • | 1 | 1 | • | • | • | • |
| M | 1 | • | • | -1 | 1 | -1 | -1 | • |
| F | 1 | 1 | • | -2 | • | -1 | -1 | 1 |
| P | • | -1 | -2 | -2 | -1 | 3 | • | -1 |
| S | • | • | • | 1 | • | • | 1 | • |
| T | -1 | -1 | -1 | • | -1 | -1 | • | -1 |
| W | 1 | • | -2 | -2 | • | -1 | -2 | -1 |
| Y | 1 | • | -1 | -1 | 1 | -1 | • | • |
| V | -1 | • | -1 | -1 | -1 | -1 | -1 | • |



36

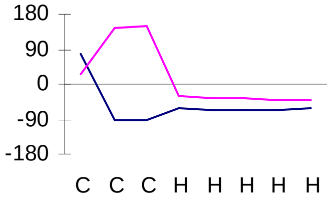


|   | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| A | • | 1 | 1 | -1 | -1 | -1 | • | • |
| R | 1 | • | • | -1 | • | -1 | • | 1 |
| N | • | • | • | 1 | -2 | 1 | • | • |
| D | • | • | -1 | • | -2 | 2 | -1 | 1 |
| C | -2 | -2 | -2 | -2 | • | -2 | -1 | -1 |
| Q | • | 1 | • | • | • | -2 | -1 | • |
| E | 1 | 1 | • | -1 | -1 | • | • | 1 |
| G | -2 | -1 | -1 | 3 | -2 | • | -1 | -1 |
| H | • | • | 1 | -1 | -1 | -1 | • | • |
| I | -1 | -1 | -1 | -3 | 1 | -3 | • | -2 |
| L | -1 | -1 | 1 | -2 | 1 | -2 | 1 | -1 |
| K | 2 | 1 | • | • | • | • | • | 1 |
| M | -1 | -1 | 1 | -1 | 1 | -2 | • | -1 |
| F | -2 | -2 | 1 | -2 | 1 | -1 | 1 | -2 |
| P | -2 | -2 | -4 | -2 | • | 1 | 2 | • |
| S | • | • | -1 | -1 | -2 | 1 | -1 | • |
| T | -1 | • | • | -3 | -1 | 1 | -1 | -1 |
| W | -1 | • | -1 | -1 | 1 | -1 | • | -1 |
| Y | -1 | -2 | 1 | -2 | 1 | -1 | • | -1 |
| V | -1 | -1 | -2 | -3 | 1 | -2 | • | -1 |

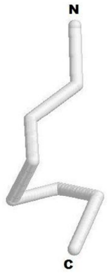


## Helix N-Caps

33



|   | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| A | -1 | • | -1 | • | • | • | 1 | 1 |
| R | -1 | • | -1 | -1 | • | • | • | • |
| N | 1 | -1 | 1 | -1 | • | • | -3 | -1 |
| D | 1 | -1 | 1 | • | 1 | 1 | -2 | -1 |
| C | -2 | • | -2 | -1 | -1 | -3 | • | • |
| Q | • | • | -1 | • | • | 1 | -1 | • |
| E | • | • | -1 | • | 2 | 1 | • | • |
| G | 2 | -2 | • | -1 | -1 | -1 | -2 | -2 |
| H | • | -1 | -1 | -1 | -1 | • | -2 | -1 |
| I | -2 | • | -4 | • | -2 | -1 | 1 | • |
| L | -2 | 1 | -2 | 1 | -1 | • | 1 | • |
| K | • | • | • | -1 | • | • | -1 | • |
| M | -1 | 1 | -2 | • | -2 | -1 | • | • |
| F | -2 | • | -2 | • | -2 | -1 | • | • |
| P | -1 | • | 1 | 1 | • | -2 | -3 | -1 |
| S | • | -1 | 2 | -1 | 1 | • | -1 | -1 |
| T | -1 | • | 1 | -1 | • | • | -1 | -1 |
| W | -2 | • | -3 | • | -1 | -1 | • | • |
| Y | -1 | 1 | -2 | • | -1 | -1 | • | • |
| V | -2 | • | -2 | • | -1 | -1 | 1 | • |



# Appendix B

# Listing of Protein Data Sets

This appendix lists the proteins found in all data sets used in this study, which include BB55, RS126, PP187, and CB396. Each protein is represented by its PDB ID and chain ID (if exist). Note that there are only 317 proteins in CB396 as some of the proteins have been split into multiple disjoint peptides to make up a total of 396 entries.

**BB55 selected by Bystroff and Baker [20]:**

```
1ANV     1APY A   1AYL     1BMF A   1BMF D   1BMF G   1BRO A   1CEM
1CPO     1DEK A   1DIV     1FIE A   1FRV A   1FRV B   1GAL     1GND
1GPL     1GTM A   1HAV A   1HLR A   1HTP     1HTT A   1HXP A   1IGN A
1IHF B   1KXU     1LBD     1LBU     1LCL     1LNH     1MSP A   1OTG A
1OXY     1QBA     1REQ A   1RIE     1SFE     1STM A   1TAQ     1TFE
1TFR     1VCC     1VHI A   1VNC     1WHI     1XEL     1XSM     1XVA A
1ZYM A   2AYH     2EBN     2ENG     2STV     4KBP A
```

**RS126 selected by Rost and Sander [37]:**

```
1A45     1ACX     1AZU     1BBP A   1BDS     1BKS A   1BKS B   1BMV 1
1BMV 2   1CBH     1CC5     1CDT A   1CRN     1CSE I   1CYO     1DUR A
1ECA     1ETU     1FC2 C   1FDL H   1FKF     1FND     1FXI A   1G6N A
```

```
1GD1 O   1GDJ     1GP1 A   1HIP     1IL8 A   1IQZ A   1L58     1LAP
1LMB 3   1MCP L   1MRT     1OVO A   1PAZ     1PPT     1PYP     1R09 2
1RBP     1RHD     1S01     1SH1     1TGS I   1TNF A   1UBQ     256B A
2AAT     2AK3 A   2ALP     2CAB     2CCY A   2CYP     2FOX     2GBP
2GLS A   2GN5     2HMZ A   2I1B     2LHB     2LTN A   2LTN B   2MEV 4
2MHU     2OR1 L   2PAB A   2PCY     2PHH     2RSP A   2SNS     2SOD B
2STV     2TGP I   2TMV P   2TSC A   2UTG A   2WRP R   3AIT     3BLM
3CD4     3CLA     3CLN     3EBX     3HMG A   3HMG B   3ICB     3PGM
3RNT     3TIM A   4BP2     4CMS     4CPA I   4CPV     4GR1     4PFK
4RHV 1   4RHV 3   4RHV 4   4RXN     4SDH A   4SGB I   4TS1 A   4XIA A
5CYT R   5ER2 E   5HVP A   5LDH     5LYZ     6ACN     6CPA     6CPP
6CTS     6DFR     6HIR     6TMN E   7CAT A   7ICD     7RSA     8ABP
8ADH     9API A   9API B   9INS B   9PAP     9WGA A
```

**PP187 selected by Jones [14]:**

```
1A34 A   1ACI     1AE9 A   1AFW B   1AH7     1AJZ     1AK0     1ALV A
1AMM     1AMU A   1AOH B   1AOP     1AOZ A   1ARS     1ARU     1AT0
1AVM A   1AYL     1BFD     1BGF     1BQU B   1CAA     1CBN     1CEI
1CEL A   1CEM     1CHM A   1CLC     1CMB A   1COY     1CPO     1CSH
1CUK     1CYN A   1CYO     1DAA A   1DJA     1DMB     1DMR     1DUP A
1ECL     1EMA     1ESF A   1EXT A   1EZM     1FKF     1FLE I   1FMK
1FUA     1FVK A   1GAI     1GD1 O   1GLQ A   1GND     1GOF     1GPB
1GPR     1GZI     1HAN     1HCZ     1HFC     1HPM     1HRD A   1HSB A
1HTR P   1HXN     1HXP A   1HYP     1IGD     1IOW     1ISO     1ISU A
1JBC     1JDW     1KAP P   1KID     1KNB     1KPT A   1KVD A   1LAM
1LDG     1LIS     1LMB 3   1LTS A   1MDL     1MLA     1MML     1MOL A
1MRK     1MSK     1MTY D   1MTY G   1MUG A   1NAH     1NNC     1NOX
1NP4     1OBW B   1OIS     1ONC     1ONR A   1OPC     1ORC     1OSP O
1OTF A   1PBE     1PGS     1PK4     1PMI     1PNK A   1PNK B   1PPN
1PTY     1QBA     1QNF     1RA9     1REG X   1RHS     1RIE     1RKD
1RPO     1RSS     1SFT A   1SGP I   1SJU     1SKZ     1SLU A   1SRI A
1STM A   1SVB     1TFE     1THG     1THV     1TVD A   1TX4 A   1TYS
1TYV     1UBS B   1UCH     1UDG     1UTG     1UXY     1VCC     1VHB A
1VHH     1VIE     1VJS     1VOM     1VPS A   1VPT     1WBA     1WER
1WHI     1WJD B   1XIK A   1YGE     1YTB A   1ZNB A   2ABK     2ARC A
2BAA     2CBA     2CCY A   2CMD     2CTC     2CY3     2END     2ENG
2ERL     2ILK     2LTN B   2MSB A   2NLL B   2OHX A   2PHY     2PSP A
2RAN     2RN2     2SIC I   2TGI     2VPF B   3CLA     3PTE     4BCL
4RHN     5CYT R   8RUC K
```

**CB396 selected by Cuff and Barton [46]:**

```
154L     1AAZ B   1ADD     1ADE B   1AHB     1ALK B   1AMG     1AMP
1AOR B   1AOZ B   1ASW     1ATP I   1AVH B   1AYA B   1BAM     1BCX
```

```
1BDO     1BET     1BFG     1BNC B   1BOV B   1BPH A   1BRS E   1BSD B
1CBG     1CDL G   1CEI     1CEL B   1CEM     1CEO     1CEW I   1CFB
1CFR     1CGU     1CHB E   1CHD     1CHK B   1CHM B   1CKS C   1CLC
1CNS B   1COI     1COL B   1COM C   1CPC L   1CPN     1CQA     1CSM B
1CTF     1CTH B   1CTM     1CTN     1CTU     1CXS A   1CYX     1DAA B
1DAR     1DEL B   1DFJ I   1DFN B   1DIH     1DIK     1DIN     1DKZ A
1DLC     1DNP B   1DPG B   1DSB B   1DTS     1DUP A   1DYN B   1ECE B
1ECL     1ECP F   1EDD     1EDM C   1EDN     1EFT     1EFU D   1EPB B
1ESE     1ESL     1EUU     1FBA B   1FBL     1FDT     1FIN D   1FJM B
1FUA     1FUQ B   1GAL     1GCB     1GCM C   1GEP     1GFL B   1GHS B
1GKY     1GLN     1GMP B   1GND     1GOG     1GP2 A   1GP2 G   1GPC
1GPM D   1GRJ     1GTM C   1GTQ B   1GYM     1HAN     1HCG B   1HCR A
1HIW S   1HJR D   1HMP B   1HMY     1HNF     1HOR B   1HPL B   1HSL B
1HTR P   1HUP     1HVQ     1HXN     1HYP     1IGN B   1ILK     1INP
1IRK     1ISA B   1ISU B   1JUD     1KIN B   1KNB     1KPT B   1KRC A
1KRC B   1KTE     1KTQ     1KUH     1LAT B   1LBA     1LBU     1LEH B
1LIB     1LIS     1LKI     1LPB A   1LPE     1MAI     1MAS B   1MCT I
1MDA J   1MDA M   1MDT A   1MJC     1MLA     1MMO H   1MNS     1MOF
1MRR B   1MSP B   1NAL 4   1NAR     1NBA C   1NCG     1NDH     1NFP
1NGA     1NLK L   1NOL     1NOX     1NOZ B   1OAC B   1ONR B   1OTG C
1OVB     1OXY     1OYC     1PBP     1PBW B   1PDA     1PDN C   1PDO
1PGA     1PHT     1PII     1PKY C   1PMI     1PNM B   1PNT     1POC
1POW B   1PPI     1PTR     1PTX     1PYT A   1QBB     1QRD B   1REC
1REG Y   1REQ C   1RHG C   1RIE     1RIS     1RLD S   1RLR     1RPO
1RSY     1RVV Z   1SCU D   1SCU E   1SEI B   1SES A   1SFE     1SFT B
1SMN B   1SMP I   1SPB P   1SRA     1SRJ A   1STF I   1STM E   1SVB
1TAB I   1TAQ     1TCB A   1TCR A   1TFR     1THT B   1THX     1TIE
1TIF     1TIG     1TII C   1TML     1TND B   1TPL B   1TRB     1TRH
1TRK B   1TSP     1TSS B   1TUL     1TUP C   1UBD C   1UDH     1UMU B
1VCA B   1VCC     1VHH     1VHR B   1VID     1VJS     1VMO B   1VNC
1VOK B   1VPT     1WAP V   1WFB B   1WHI     1XVA B   1YPT B   1YRN A
1ZNB B   1ZYM B   2AAI B   2ABK     2ADM B   2AFN C   2ASR     2BAT
2BLT B   2BOP A   2CMD     2CPO     2DKB     2DLN     2DNJ A   2EBN
2END     2ERL     2GSQ     2HFT     2HHM B   2HIP B   2HPR     2MLT B
2MTA C   2NAD B   2NPX     2OLB A   2PGD     2PHY     2POL B   2REB
2RSL A   2SCP B   2SIL     2SPT     2TGI     2TMD B   2TRT     2YHX
3BCL     3CHY     3COX     3ECA B   3INK D   3MDD B   3PGK     3PMG B
4FIS B   5SIC I   6RLX C   6RLX D   821P
```

# Bibliography

[1] C. B. Anfinsen. *Principles that govern the folding of protein chains*, Science, 181:223-230, 1973

[2] X. Yuan, Y. Shao, C. Bystroff. *Ab initio protein structure prediction using pathway models*, Comparative and Functional Genomics, 4(4):397-401, 2003

[3] C. Bystroff, K. T. Simons, K. F. Han, D. Baker. *Local sequence-structure correlations in proteins*, Current Opinion in Biotechnology, 7:417-421, 1996

[4] J. Xu, M. Li, D. Kim, Y. Xu. *Optimal protein threading by linear programming*, Journal of Bioinformatics and Computational Biology, 1(1):95-117, 2003

[5] W. A. Hendrickson. *Stereochemically Restrained Refinement of Macromolecular Structures*, Methods in Enzymology, 115:252-270, 1985

[6] M. S. Weiss, A. Jabs, R. Hilgenfeld. *Peptide bonds revisited*, Nature Structural & Molecular Biology, 5(8):676, 1998

[7] C. Sander, R. Schneider. *Database of homology-derived protein structures and the structural meaning of sequence alignments*, Proteins, 9:56-68, 1991

[8] S. F. Altschul, T. L. Madden, A. A. Schaffer, J. Zhang, Z. Zhang, W. Miller, D. J. Lipman. *Gapped Blast and PSI-Blast: a new generation of protein database search programs*, Nucleic Acids Research, 25(17):3389-402, 1997

[9] P. Sibbald, P. Argos. *Weighting aligned protein or nucleic acid sequences to correct for unequal representation*, Journal of Molecular Biology, 216:813-818, 1990

[10] S. R. Eddy, G. Mitchison, R. Durbin. *Maximum discrimination hidden Markov models of sequence consensus*, Journal of Computational Biology, 2:9-23, 1995

[11] S. Henikoff, J. G. Henikoff. *Position-based sequence weights*. Journal of Molecular Biology, 243:574-578, 1994

[12] B. Rost, C. Sander. *Improved prediction of protein secondary structure by use of sequence profiles and neural networks*, Proceedings of the National Academy of Sciences USA, 90:7558-7562, 1993

[13] G. de Brevern, C. Etchebest, S. Hazout. *Bayesian probabilistic approach for predicting backbone structures in terms of protein blocks*, Proteins, 41:271-287, 2000

[14] D. T. Jones. *Protein secondary structure prediction based on position-specific scoring matrices*, Journal of Molecular Biology, 292:195-202, 1999

[15] J. Wojcik, J. Mornon, J. Chomilier. *New efficient statistical sequence-dependent structure prediction of short to medium-sized protein loops based on an exhaustive loop classification*, Journal of Molecular Biology, 289:1469-1490, 1999

[16] R. Kolodny, P. Koehl, L. Guibas, M. Levitt. *Small libraries of protein fragments model nature protein structures accurately*, Journal of Molecular Biology, 323:297-307, 2002

[17] C. G. Hunter, S. Subramaniam. *Protein fragment clustering and canonical local shapes*, Proteins, 50:580-588, 2003

[18] J. MacQueen. *Some methods for classification and analysis of multivariate observations*, Proceedings of the 5th Berkeley Symposium, 1:281-297, 1967

[19] B. S. Everitt. *Cluster Analysis 3rd Edition*, Halsted Press, New York, 1993

[20] C. Bystroff, D. Baker. *Prediction of local structure in proteins using a library of sequence-structure motifs*, Journal of Molecular Biology, 281:565-577, 1998

[21] A. Yang, L. Wang. *Local structure prediction with local structure-based sequence profiles*, Bioinformatics, 19(10):1267-1274, 2003

[22] B. Oliva, P. A. Bates, E. Querol, F. X. Aviles, M. J. Sternberg. *An automated classification of the structure of protein loops*, Journal of Molecular Biology, 266:814-830, 1997

[23] R. L. Baldwin, G. D. Rose. *Is protein folding hierarchic? II. Folding intermediates and transition states*, Trends in Biochemical. Sciences, 24:77-83

[24] C. Schellman. The αL conformation at the ends of helices, Protein Folding: Proceedings of the 28th Conference of the German Biochemical Society, 1980:53-56, 1979

[25] V. Munoz, F. J. Blanco, L. Serrano. *The hydrophobic-staple motif and a role for loop residues in α-helix stability and protein folding*, Nature Structural & Molecular Biology, 2:380-385, 1995

[26] E. T. Harper, G. D. Rose. *Helix stop signals in proteins and peptides: the capping box*, Biochemistry, 32:7605-7609, 1993

[27] F. J. Blanco, L. Serrano. *A short linear peptide that folds into a native stable beta-hairpin in aqueous solution*, Nature Structural & Molecular Biology, 1:584-590, 1994

[28] E. de Alba, M. A. Jimenez, M. Rico, J. L. Nieto. *Conformational investigation of designed short linear peptides able to fold into β-hairpin structures in aqueous solution*, Folding & Design, 1:133-144, 1996

[29] R. Bonneau, J. Tsai, I. Ruczinski, D. Chivian, C. Rohl, C. E. Strauss, D. Baker. *Rosetta in CASP4: progress in ab initio protein structure prediction*, Proteins, Supplement 5, 119-126, 2001

[30] K. Fidelis, P. S. Stern, D. Bacon, J. Moult. *Comparison of systematic search and database methods for constructing segments of protein structure*, Protein Engineering, 7:953-960, 1994

[31] M. J. Rooman, J. P. Kocher, S. J. Wodak. *Prediction of backbone conformation based on seven structural assignments. Influence of local interactions*, Journal of Molecular Biology, 221:961-979, 1991

[32] R. L. Tatusov, S. F. Altschul, E. V. Koonin. *Detection of conserved segments in proteins: iterative scanning of sequence databases with alignment blocks*, Proceedings of the National Academy of Sciences USA, 91:12091-12095, 1994

[33] R. B. Russell, R. R. Copley, G. J. Barton. *Protein fold recognition by mapping predicted secondary structures*, Journal of Molecular Biology, 259:349-365, 1996

[34] P. Baldi, S. Brunak, P. Frasconi, G. Pollastri, G. Soda. *Exploiting the Past and the Future in Protein Secondary Structure Prediction*, Bioinformatics, 15:937-946, 1999

[35] R. D. King, M. J. Sternberg. *Identification and application of the concepts important for accurate and reliable protein secondary structure prediction*, Protein Science, 5(11):2298-2310, 1996

[36] T. M Yi, S. Lander. *Protein secondary structure prediction using nearest-neighbor methods*, Journal of Molecular Biology, 232:1117-1129, 1993

[37] B. Rost, C. Sander. *Prediction of secondary structure at better than 70% accuracy,* Journal of Molecular Biology, 232:584-599, 1993

[38] D. Frishman, P. Argos. *Incorporation of non-local interactions in protein secondary structure prediction from the amino acid sequence*, Protein Engineering, 9(2):133-142, 1996

[39] S. Hua, Z. Sun. *A novel method of protein secondary structure prediction with high segment overlap measure: support vector machine approach*, Journal of Molecular Biology, 308:397-407, 2001

[40] M. J. J. M. Zvelebil, G. J. Barton, W. R. Taylor, and M. J. Sternberg. *Prediction of protein secondary structure and active sites using the alignment of homologous sequences*, Journal of Molecular Biology, 195:957-961, 1987

[41] J. Guo, H. Chen, Z. Sun, Y. Lin. *A novel method for protein secondary structure prediction using dual-layer SVM and profiles*. Proteins, 54(4):738-743, 2004

[42] C. Burges. *A tutorial on support vector machines for pattern recognition*, Data Mining and Knowledge Discovery 2, 121-167, 1998

[43] R. A. Engh, R. Huber. *Accurate bond and angle parameters for X-ray protein structure refinement*, Acta Crystallographica, A47, 392-400, 1991

[44] R. Kolodny, P. Koehl, L. Guibas, M. Levitt. *Small libraries of protein fragments model nature protein structures accurately*, Journal of Molecular Biology, 323:297-307, 2002

[45] U. Hobohom, M. Scharf, R. Schneider, C. Sander. *Selection of representative protein data sets*, Protein, 1:409-417, 1992

[46] U. Hobohom, C. Sander. *PDB Select 25, July 03 (http://homepages.fh -giessen.de/)*

[47] W. Kabach, C. Sander. *A dictionary of protein secondary structure*, Biopolymers, 22:2577-2637, 1983

[48] M. R. Anderberg. *Cluster analysis for applications*, Academic Press, New York, 1973

[49] J. A. Cuff, G. J. Barton. *Evaluation and improvement of multiple sequence methods for protein secondary structure prediction*, Proteins, 34:508-519, 1999

[50] C. Bystroff. I-sites Library (*http://www.bioinfo.rpi.edu/application/i-sites/Isites/*)

[51] M. Lesk. *CASP2: report on ab initio predictions*, Proteins, Supplement 1, 151-166, 1997

[52] S. Umeyama. *Least squares estimation of transformation parameters between two point patterns*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 13(4):376-386, 1991

[53] N. Qian, T. J. Sejnowski. *Predicting the secondary structure of globular proteins using neural network models*, Journal of Molecular Biology, 202:865-884, 1988

[54] B. W. Matthews. *Comparison of the predicted and observed secondary structure of T4 phage lysozyme*, Biochimica et Biophysica Acta., 405:442-451, 1975

[55] B. Rost, C. Sander, R. Schneider. *Redefining the goals of protein secondary structure prediction*, Journal of Molecular Biology, 235:13-26, 1994

[56] A. Zemla, C. Venclovas, K. Fidelis, B. Rost. *A modified definition of sov, a segment-based measure for protein secondary structure prediction assessment*, Proteins, 34:220-223, 1999

[57] T. Joachims, *Making large-Scale SVM Learning Practical*. Advances in Kernel Methods - Support Vector Learning, B. Schölkopf and C. Burges and A. Smola (ed.), MIT-Press, 1999