Pattern Mining of Protein Contact Networks

A thesis submitted to the University of Hyderabad in partial fulfillment of the requirements for the award of

Doctor of Philosophy
in
Computer Science

by

Suvarna Vani Koneru

07MCPC10

School of Computer & Information Sciences
University of Hyderabad
Hyderabad – 500 046, India

June 2014

CERTIFICATE

This is to certify that the thesis entitled **Pattern Mining of Protein Contact Networks** submitted by **Koneru Suvarna Vani** bearing Reg. No. **07MCPC10** in partial fulfillment of the requirements for the award of **Doctor of Philosophy** in **Computer Science** is a bonafide work carried out by her under my supervision and guidance.

This thesis has not been submitted previously in part or in full to this or any other University or Institution for the award of any degree or diploma.

Dr. S. Durga Bhavani
Supervisor
School of Computer & Information Sciences
University of Hyderabad
Hyderabad -500 046

Prof. A. K. Pujari
Dean
School of Computer & Information Sciences
University of Hyderabad
Hyderabad -500 046

DEDICATION

Dedicated

To

My Family

DECLARATION

I, **Koneru Suvarna Vani**, hereby declare that this thesis entitled **Pattern Mining of Protein Contact Networks** submitted by me under the guidance and supervision of **Dr. S. Durga Bhavani** is a bonafide research work. I also declare that it has not been submitted previously in part or in full to this University or any other University or Institution for the award of any degree or diploma.

Date K Suvarna Vani

Place

Acknowledgments

It is my pleasure to thank all the people who have helped me along the way for completion of Ph.D. In the first order, I would like to express my profuse thanks to my supervisor Dr. S. Durga Bhavani for introducing me to the field of Bioinformatics. I value the interesting discussions we had, taught me the practical aspects of research and I feel that I am truly privileged to work under her supervision. I am grateful to her for being kind, patient and encouraging me through out the course.

I am extremely grateful to our Dean, School of Computer Information Sciences, Prof. Arun K. Pujari and previous HODs for providing excellent computing facilities and a nice working atmosphere.

My sincere thanks to Dr. Somdatta Sinha, Center for Cellular and Molecular Biology(CCMB), Hyderabad for introducing me to the field of Bioinformatics, also I thank Pankaj Barah for sharing data sets in this domain. I also thank Dr. Ram Rupsarkar advisor of Summer Research Fellowship Programme(SRFP2010).

My sincere thanks to my doctoral committee members Dr. C. R. Rao and Dr. S. K. Udgata for their guidance through out my course. I thank Dr. Chakravarthy Bhagavathi, Dr. Atul Negi and Dr. Anupama Potluri for their valuable suggestions. A special thanks to Dr. Bapi Raju and Dr. T. Shoba Rani for their advice and support giving valuble comments on my draft.

My heartfelt thanks to my husband Praveen Kumar Kollu for his complete support and encouragement in through out my course. I also thank my kids Sree Varshitha Kollu, Sai Venkata Vignesh Kollu. I thank my sister-in-law Neerja Kollu, brother Srinivasa Rao Talluri, Sachindra Talluri and Sindhuja Talluri.

My heartfelt thanks to my parents and In-laws who have supported in every aspect of my education and carrer and for their struggle beyond their potential. I thank my father for his struggle to educate us. I thank my mother for her love, care and encouragement.

My sincere thanks to our HoDs Dr. K. V. Sambasiva Rao, Dr. Sushma Yalamanchili, Mr. G. Krishna Kishore and Dr. V. Srinivasa Rao for leaves sanctioned. My heartfelt thanks to our college principals Dr. K. R. K. Prasad garu who motivated me to pursue Ph.D, Dr.

K. Mohana Rao garu who sanctioned leave for Summer Research Fellowship Program (SRFP10) and encouraged me and Dr. G. Sambasiva Rao garu sanctioned leaves whenever required and sanctioned funding for interantional travel grant for IEEE IJCNN 2013, University of Arlington, Texas, USA. I thank Dr. G. N. Swamy, TEQIP-II co-ordinator, Katragadda Raghu and team for their support of international travel.

My heartfelt thanks to Siddhartha Academy of General and Technical Education(SAGTE), Vijayawada. Especially my thanks to our seceratarie Sri P. L. N. Prasad garu, Sri Nalluri Venkateswarlu garu and Sri Paladugu Lakshmana Rao garu, convener of our college Sri. Myneni Rajayya garu and all life members of our academy.

Date K Suvarna Vani

Abstract

The thesis proposes an alternate way of solving challenging problems of computational biology like protein secondary structure prediction, protein fold recognition, protein fold signatures and contact map overlap problem by exploiting the idea that proteins belonging to the same protein fold have 'similar' contact maps. Pattern mining of contact maps is conducted to derive features from clusters of contacts rather than clusters of nodes that are in contact. Using the work in the literature that predicts contact maps from the primary amino acid sequence, we propose that using pattern features from predicted contact maps would lead to an *ab-initio* method.

An algorithm called *Extract_SSP* based on heuristics that extract configurational statistics of secondary structure elements(SSE) of helices and beta strands from contact maps is proposed. Protein secondary structure prediction is achieved with an accuracy of 76% with a good accuracy obtained for helix prediction at 91% on par with the best results in the literature. Further, we propose an algorithm *ExtractPatterns* that extracts (non)rectangular 2D clusters of contacts from the off-diagonal region in linear time. A feature vector of length 11 is formed using this study constituting diagonal and off-diagonal statistical features.

Proteins can be classified among the four structural classes of All-Alpha, All-Beta, Alpha+Beta and Alpha/Beta which are further subdivided into 27 folds. Protein fold classification problem is cited in the literature as a challenging unbalanced classification problem with the accuracy results being as low as 51.1% on the bench mark data set of Ding et al. and highest accuracy at 60.5% using 100 features. We represent the proteins as 11-length feature vectors and adopt Synthetic Minority Over-sampling Technique(SMOTE) based boosting approach to balance the data to address the 27-way fold classification problem. C4.5 decision tree classifier is built which in combination with SMOTE boosting algorithm using the novel contact map features showed an enhanced prediction accuracy of 65.25%. An additional advantage of our approach is the reduced dimensionality of the feature vector which is 11 whereas literature uses more than 100 features on average.

'Signature' rules are derived for each protein fold using class based association rule mining algorithm. These rules give important insights into the significant 3D substructures that may be crucial for a protein to assume a specific fold conformation. Further, we demonstrate correspondence between the 2D-patterns in the contact map of a fold to the specific 3D-motifs located within the tertiary structure and its functional importance. We validate these ideas further by using the 2D-patterns as features and carry out the challenging fold recognition problem. Using class based

association rule mining algorithm these signatures achieve the highest accuracy of 71.55% on the 27-way fold classification problem.

Protein structure comparison is modeled in the literature as Contact map overlap(CMO) problem which is an important NP-Hard problem in this area. We propose a divide and conquer approach to CMO by dividing the whole contact map into smaller contact maps using *ExtractPatterns* algorithm. Using *Approximate 2D-Pattern Matching* algorithm and dynamic programming approach, we find matching between the smaller contact maps. We choose a CMO algorithm called Multistart Variable Neighborhood Search(MSVNS) to align these smaller contact maps and the results are 'merged' to obtain global alignment. The CMO is computed on a bench mark data set called *Skolnick* and an average overlap of 84% of MSVNS is obtained. On certain folds, the proposed algorithm obtained better results compared to MSVNS. Our implementations show that, on average, our algorithm takes less than one minute whereas global alignment takes more than 5 minutes to compute maximum overlap between a pair of proteins.

This work validates the hypothesis that contact maps contain useful information that can be utilized by machine learning approaches to address protein fold classification and other related problems quite effectively.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Primary structure of a protein is considered as simply a string on a 20-symbol alphabet of aminoacid residues. Anfinsen's famous experiment showing the automatic folding of a protein primary sequence *invivo* without any external stimulus [13] lead the researchers to believe that the entire structure information is embedded within the primary sequence of the protein. Protein structure prediction ab-initio, that is from the knowledge of its constituent amino acids, is considered as one of the main challenges for researchers in the bioinformatics community.

Protein contact map is a 2-dimensional matrix representation of the protein 3D structure. A protein contact map is a binary adjacency matrix A of order $n \times n$ where n is the number of amino acids in the protein, with $A(i, j) = 1$ if $d(i, j) < t$ and 0 otherwise, with 'd' being euclidean distance and t conventionally being chosen between 6 and 10 Angstrom units [84]. Figure 1.1 shows 3D structure of protein 2IGD of length 61, cartoon topology and the corresponding contact map.

Protein structure prediction that uses contact maps as intermediaires is conducted generally in two steps: firstly, predict the contact map from the primary sequence and secondly, predict the 3D-structure from the predicted contact maps. Many machine-learning methods have been developed for the first step that of protein contact map prediction [82, 32, 53, 16] which uses amino acid features and classifiers like Support vector machines and Neural networks for predicting the presence or absence of a contact. The second step of predicting the protein structure from the contact map, that is recovering a set of 3D coordinates consistent with the given contact map is equivalent to unit-disk-graph realization problem which has been proved to be NP-Hard [18]. Many heurisitc algorithms have been proposed for protein structure reconstruction [81, 84].

Our work addresses the second problem in an indirect fashion. We do not retrieve the

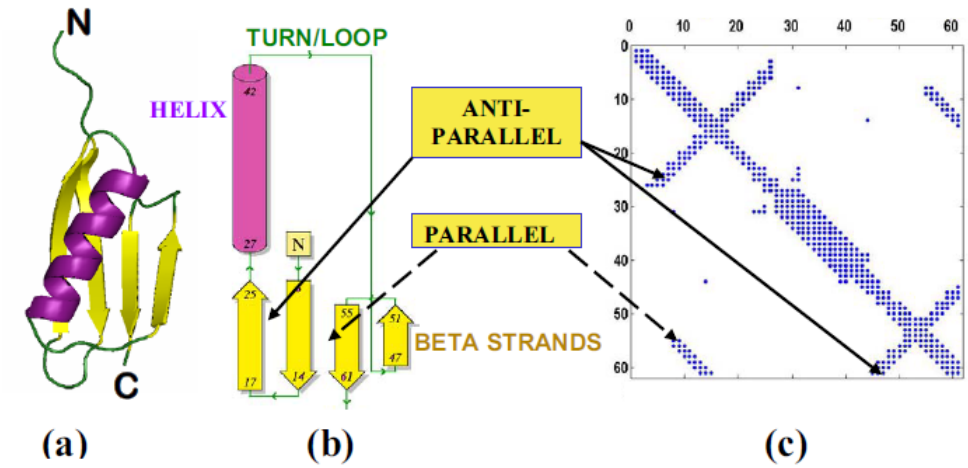


Figure 1.1: 3D Structure and cartoon topology for Protein 2IGD [14]

3D coordinates but solve a sub-problem of protein structure prediction, namely protein fold classification problem. It is believed that all the possible structures of a protein may fall in among approximately 1000 fold classes [2]. Hence it is an interesting problem to predict the structural class and the fold of a protein. Structural Classification of Proteins (SCOP) [2] identifies four major structural classes for protein structures, viz All-Alpha, All-Beta, Alpha/Beta and Alpha+Beta. Proteins within a class are further grouped into 'folds', those containing a specific combination of major secondary structural elements connected in a similar topological configuration.

Since protein contact maps can be predicted ab-initio, why can't we use 2D-pattern features from the predicted contact map in a purely computational way and use machine learning methods to solve the challenging problems like protein fold recognition and protein structure comparison. This would provide an alternate 'ab-initio' method which is purely computational, without resorting to domain biological knowledge, and that solves the structure classification problems.

1.1 Motivation

Somdatta et al. were the first to indicate that contact map analysis could be directly used for fold discrimination [14]. They demonstrate in a few examples that proteins belonging to the same fold have 'similar' looking contact maps. Visually, a few distinguishing patterns are visible for each fold. We work on building automated procedures that derive these patterns corresponding to each fold. Hence if a protein sequence with unknown structure is given,

we obtain the contact map of the protein using the prediction tools available [3], and then using our automated system predict the structural class and fold that the protein belongs to.

Our work mainly focuses on proposing an alternate route to solve classical problems of computational biology like protein secondary structure prediction, protein fold recognition, protein fold signatures and contact map overlap problem by mining contact maps and using machine learning approaches.

1.2 Contributions

- We propose a completely novel approach to solve the protein fold recognition problem using data mining of contact maps.
- We propose 2D-pattern extraction algorithms that are effective for sparse binary matrices.
- Satisfactory results have been obtained for protein secondary structure prediction using simple heuristics.
- Using association rule mining, we derive 'signature' rule for a protein fold. This approach gives immense insights into the significant 3D substructures that may be crucial for a protein to assume a specific fold conformation.
- We propose a novel algorithm to address the NP-Hard problem of contact map overlap problem using the insights gained. The proposed algorithm is a parallel algorithm and hence reduces the computation time. Moreover, the alignment is built based on sub-structure alignment and hence more structurally meaningful.

1.3 Thesis Organization

Chapter 1 gives a brief introduction to the problems addressed and also the outlay of the chapters.

Chapter 2 discusses the preliminaries from bioinformatics that are required for a computer science reader.

Chapter 3 onwards carries the main body of the work focussing on feature extraction from the contact maps. Contact map is a sparse binary matrix with contacts depicted as 1's. Majority of the contacts in the matrix are present in the diagonal region representing short-range interactions between the secondary structures and one can see islands of contacts in

the off-diagonal region corresponding to long-range interactions between the amino acid residues. In **Chapter 3**, we propose algorithms that extract features from the diagonal region as well as the off-diagonal region of the contact map. Using simple heursitics, protein secondary structures are extracted successfully from the diagonal information. We compare the results obtained by our algorithm to the existing algorithms on the bench-mark data set RS126. Further, in order to validate the approach, the algorithm is tested on predicted contact maps. In addition, connected regions of contacts are extracted from the off-diagonal region using *ExtractPatterns* algorithm. As part of a typical machine learning approach, features from the diagonal region as well as off-diagonal region are composed to form a feature vector.

Chapter 4 addresses the protein fold classification problem which is a multi-class classification problem with bench-mark data sets available being highly unbalanced. A protein is represented as a 11-length feature vector constituting secondary structure features and pattern features that have been derived in the previous chapter. Very few researchers applied boosting techniques to solve this classification problem. We adopt SMOTE based boosting approach and apply J48 decision tree classifier [4] and show that a significant increase in accuracy is achieved for this challenging classification problem in comparison to the current algorithms available in the literature.

In **Chapter 5**, 'Protein fold signatures' are extracted using class-based association rule mining algorithm for the folds. Turcotte et al. [80] propose rules for fold prediction based on Inductive Logic Programming (ILP) using features pertaining to the protein 3D-structure. We derive a non-redundant set of conserved patterns from the protein contact map using *Sig_Pattern* algorithm. We show that the patterns thus found correspond to significant 3D motifs of a protein fold. Using Frequent item set mining [41] and Class-based association rule-mining algorithm [54], rules are obtained for each fold of a protein structural class. Then these rules are tested on empirically predicted contact maps to identify folds.

Based on these insights, as a culmination of our ideas, in **Chapter 6**, we propose a divide and conquer approach to address Contact Map Overlap(CMO) problem which is a well-known NP-Hard problem in this area. A scoring matrix is computed between the different regions of the two contact maps using *Approx-2D-Pattern matching* algorithm. Then by applying Dynamic programming algorithm, pairs of smaller matching contact maps are obtained. Between each pair of aligned regions an existing CMO algorithm like MSVNS [60] or Bimal [19], is applied to compute the alignment at the residue level. These local alignments are 'merged' in order to obtain a global alignment.

The divide and conquer approach facilitates parallel implementation and hence the time taken for the alignment is effectively equal to the maximum time taken by the local alignments. Our implementations show that, on average, this takes about 50 seconds whereas global alignment takes more than 5 minutes between a pair of proteins.

Chapter 2

Preliminaries of Protein Contact Networks

Protein structure prediction, a challenging problem in bioinformatics, is to predict the 3D structure of a protein from its primary structure, which is simply a finite string of amino acid residues. The famous Anfinsen's hypothesis claims that protein structure (folding) information fully resides in the corresponding primary structure of the protein and hence theoretically prediction is possible.

2.1 Protein structure

2.1.1 Primary sequence

Primary structure of a protein refers to the linear sequence of amino acids that make up the polypeptide chain. Adjacent amino acid residues are connected by a peptide bond which forms the backbone of a protein.

2.1.2 Secondary structure elements (SSE)

Repetition of a regular pattern of 'twists','turns' and 'coils' of the polypeptide chain leads to the formation of substructures within the chain which are referred to as the secondary structures of protein molecule. We consider mainly three types of secondary structures namely, alpha helix, beta sheet and turn/coil depicted in Figure2.1. Hence at the secondary structure level, the protein structure can be viewed as an annotation of primary sequence

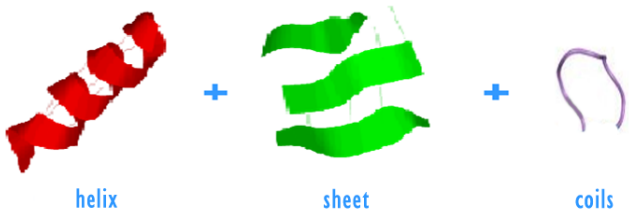


Figure 2.1: Secondary structure elements(SSE)

with secondary structure elements(SSE), such as α-helices(H), β-sheets(E), and turns(T) which together constitute the overall three-dimensional configuration of the protein chain.

Table 2.1: Primary sequence with secondary structure annotation

Primary sequence	E	D	P	E	V	L	F	K	N	K	G	C	V	A	C	H	A	I
SSE annotation	H	H	H	H	H	H	G	G	G	E	E	E	E	E	H	H	H	H

2.1.3 Tertiary structure

Tertiary structure refers to the three dimensional globular structure formed by bending and twisting of the secondary structural elements. The protein structure can be considered as a folding of secondary structure elements, such as α-helices and β-sheets, which together constitute the overall three-dimensional configuration of the protein chain. Protein Data Bank(PDB) gives entire information relating to the structure from primary, secondary to tertiary structures of a protein. Figure 2.2 shows an example of the different rules of structures of a protein.

Structural classification of proteins database(SCOP) is largely a manual classification of protein structural domains based on similarities of their structures and amino acid sequences. Proteins having same shape and similarity of sequence and/or function are placed in "families", and are assumed to have a closer common ancestor, whereas proteins with the same shape but having little sequence or functional similarity are placed in different "superfamilies", and are assumed to have only a very distant common ancestor.

2.2 Structural classification of proteins (SCOP) database

One of main sources for protein structures is the Protein Data Bank [5]. The unit of classification of structures in SCOP is a protein domain, and the shapes of domains are called

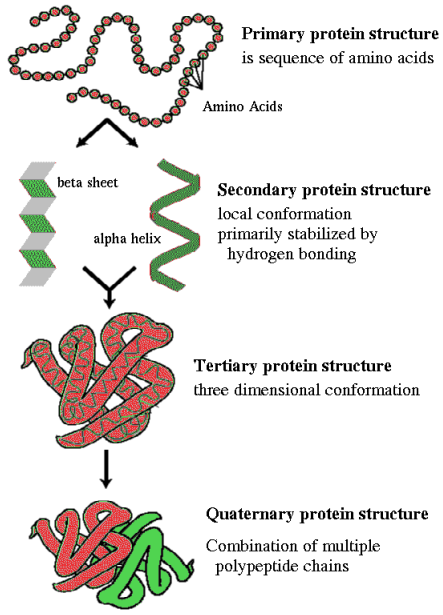


Figure 2.2: Protein hierarchical structure [1]

"folds". The SCOP [2] classification identifies four major structural classes for protein structures, viz All-Alpha, All-Beta, Alpha/Beta and Alpha+Beta. Proteins are considered to have a common fold if they contain a specific combination of major secondary structural elements having the same topological connections. A snapshot of the Fold data base is given in Table 2.2. Fold recognition or prediction, from the knowledge of its constituent amino acids, is an active area of inquiry in computational biology.

Table 2.2: SCOP classification

Class	Number of folds	Number of Superfamilies
All-Alpha proteins	126	175
All-Beta proteins	81	147
Alpha/Beta proteins (a/b)	87	135
Alpha+Beta proteins (a+b)	151	214

2.3 Protein contact networks(PCN)

Contact maps provide a reduced representation of a protein structure. The three dimensional structure of a protein can be described by a graph made from its constituent amino

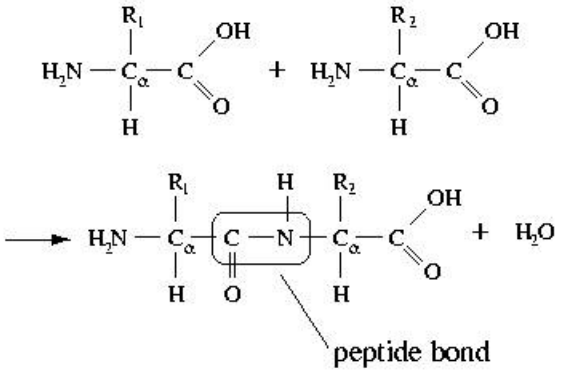


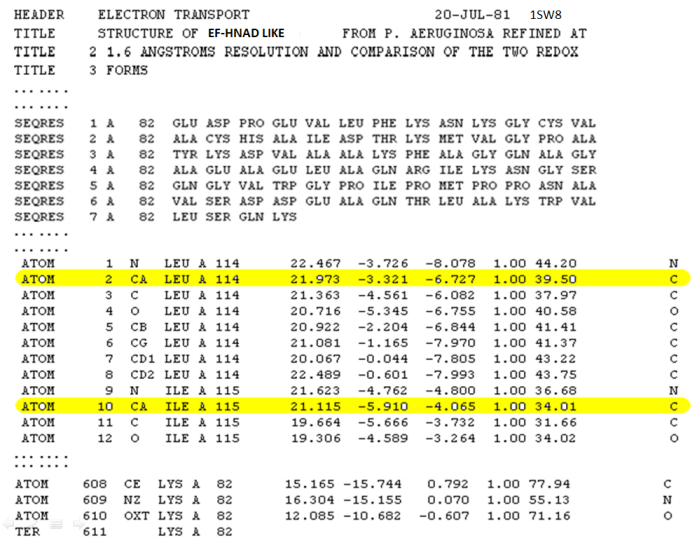
Figure 2.3: Aminoacid residues joined by a peptide bond forming the backbone

acid residues with the C_α atoms of the residues as the 'nodes', and their inter-residue interactions (covalent and non-covalent) as 'links' among the nodes. Protein contact map describes the pairwise spatial relationship of residues in a protein. Generally, only the backbone atoms included in the peptide bond are considered for the protein structure. The structure data is available as 3D coordinates of all constituent atoms of a protein in Protein Data Bank (PDB) [5]. Protein distance maps are generated from the structural data available by computing pairwise distances of all the C_α atoms. The distance map is a 2D symmetric, square matrix where the entry (i, j) represents the euclidean distance between the nodes i and j in the 3D structure. The corresponding protein contact network (PCN) of the protein of sequence length N is a boolean matrix (adjacency matrix) A of order $N \times N$, whose elements are $A(i, j)$, where $A(i, j) = 1$ if the central atoms (C_α) of the amino acid residues i and j in the protein structure are within a threshold distance, and otherwise $A(i, j) = 0$.

2.3.1 Constructing contact maps

Figure 2.3 shows the backbone of a protein, the central part indicating $C\alpha$ carbon atoms. Figure 2.4 shows the protein structure available as 3D coordinates of all its constituent atoms in the general format of pdb file. We extract the 3D-coordinates of the $C\alpha$ carbon atoms and compute the euclidean distance between all the pairs of $C\alpha$ atoms in order to obtain the distance matrix of the protein. Contact map of A is constructed with $A(x, y) = 1$ if $d(x, y) \leq t$ and 0 otherwise by setting $t = 7A^o$.

The Figure 2.5 shows the contact map of the protein 1SW8 which is constructed from the structural coordinates taken from PDB and applying threshold distance of $7A^o$. The advantage of this representation is that contact maps are invariant to rotations and translations



```
HEADER    ELECTRON TRANSPORT                      20-JUL-81   1SW8
TITLE     STRUCTURE OF EF-HNAD LIKE       FROM P. AERUGINOSA REFINED AT
TITLE    2 1.6 ANGSTROMS RESOLUTION AND COMPARISON OF THE TWO REDOX
TITLE    3 FORMS
.... ....
.... ....
SEQRES   1 A   82  GLU ASP PRO GLU VAL LEU PHE LYS ASN LYS GLY CYS VAL
SEQRES   2 A   82  ALA CYS HIS ALA ILE ASP THR LYS MET VAL GLY PRO ALA
SEQRES   3 A   82  TYR LYS ASP VAL ALA ALA LYS PHE ALA GLY GLN ALA GLY
SEQRES   4 A   82  ALA GLU ALA GLU LEU ALA GLN ARG ILE LYS ASN GLY SER
SEQRES   5 A   82  GLN GLY VAL TRP GLY PRO ILE PRO MET PRO PRO ASN ALA
SEQRES   6 A   82  VAL SER ASP ASP GLU ALA GLN THR LEU ALA LYS TRP VAL
SEQRES   7 A   82  LEU SER GLN LYS
.... ....
.... ....
ATOM      1  N   LEU A 114      22.467  -3.726  -8.078  1.00 44.20           N
ATOM      2  CA  LEU A 114      21.973  -3.321  -6.727  1.00 39.50           C
ATOM      3  C   LEU A 114      21.363  -4.561  -6.082  1.00 37.97           C
ATOM      4  O   LEU A 114      20.716  -5.345  -6.755  1.00 40.58           O
ATOM      5  CB  LEU A 114      20.922  -2.204  -6.844  1.00 41.41           C
ATOM      6  CG  LEU A 114      21.081  -1.165  -7.970  1.00 41.37           C
ATOM      7  CD1 LEU A 114      20.067  -0.044  -7.805  1.00 43.22           C
ATOM      8  CD2 LEU A 114      22.489  -0.601  -7.993  1.00 43.75           C
ATOM      9  N   ILE A 115      21.623  -4.762  -4.800  1.00 36.68           N
ATOM     10  CA  ILE A 115      21.115  -5.910  -4.065  1.00 34.01           C
ATOM     11  C   ILE A 115      19.664  -5.666  -3.732  1.00 31.66           C
ATOM     12  O   ILE A 115      19.306  -4.589  -3.264  1.00 34.02           O
::: :::
::: :::
ATOM    608  CE  LYS A  82      15.165 -15.744   0.792  1.00 77.94           C
ATOM    609  NZ  LYS A  82      16.304 -15.155   0.070  1.00 55.13           N
ATOM    610  OXT LYS A  82      12.085 -10.682  -0.607  1.00 71.16           O
TER     611      LYS A  82
```

Figure 2.4: General format of PDB structure of protein 1SW8

and can be predicted by machine learning methods [97]. It has also been shown that under certain circumstances it is possible to reconstruct the 3D coordinates of a protein using its contact map [84].

2.4 Predicted contact maps

There is immense work in the literature on prediction of protein contact map from the primary sequence of a protein. Predicting contact map using sequence information has been an active research topic in recent years since contact maps have been found to be useful for protein 3D structure prediction [59, 81, 83, 91, 100]. Protein contact map has also been used to study protein structure alignment [20, 89, 94].

Many machine-learning methods have been developed for protein contact map prediction [32, 38, 58, 62, 82, 87].SVMSEQ [92], SVMcon [24], use Support Vector Machines with features derived from sequence homologs; NNcon [76], CMAPpro [53], PSICOV [45] and Evfold [57] that predict contacts by using only amino acid sequence features derived from the primary sequence.

Distill, a software developed by Bau et al. [3] provides different servers relating to protein structure prediction problem which are all available for public use. Of these, **XXStout** is a server that predicts protein contact maps from the primary sequence. The underlying algorithms use a 2D recursive neural network [16] and provide improved versions that work in the cases when only remote homologue information is present [56]. This software has

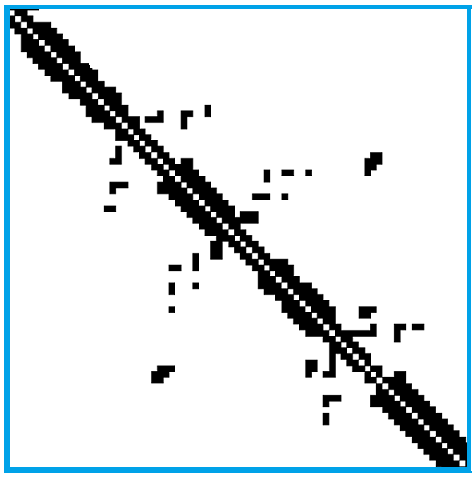


Figure 2.5: Contact map of protein 1SW8

proved to be very useful for us in building data sets of predicted contact maps.

Hence instead of extracting features from the primary amino acid sequence, we propose to extract pattern features from the predicted protein contact maps. The overall goal of the thesis is to extract as much information as possible from the predicted contact maps in order to propose alternate solutions to challenging problems like protein fold recognition and protein structure comparison. To the best of our knowledge, this kind of framework with knowledge discovery from protein contact maps to solve classification problems has not been carried out.

The overall plan of work that is carried out and being presented in the thesis is given in Figure 2.6.

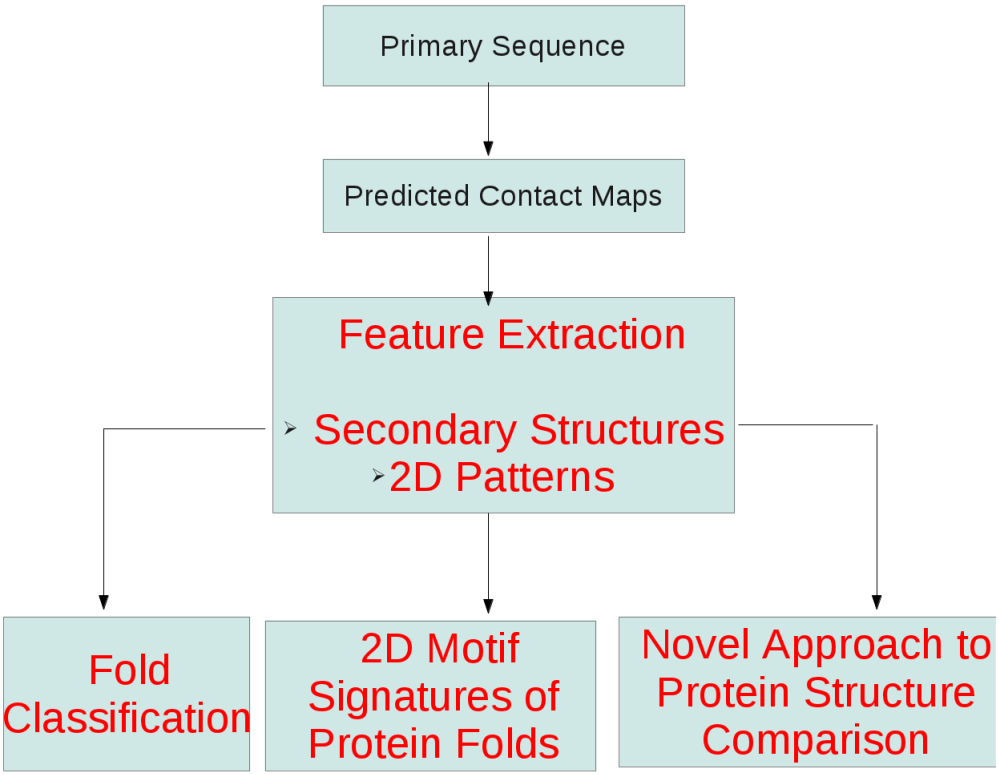


Figure 2.6: Contributions made in the thesis are from the Feature Extraction module downwards

Chapter 3

Feature Extraction of Protein Contact Matrices

3.1 Introduction

Protein contact map is a 2-dimensional representation of the protein tertiary structure. Prediction of protein contact map from the primary sequence of a protein has been addressed in the literature and software is available now to predict contact maps [3]. Hence instead of extracting features from the primary amino acid sequence, we propose to extract pattern features from the protein contact maps. We demonstrate the fact that analysis of contact maps can yield important insights for protein structure prediction. It is well known that the secondary structure elements of a protein are transparently laid out in the contact map, though no one framed rules to extract them from the contact map.

3.2 Literature

The protein secondary structure prediction problem is a well known problem in bio-informatics community for last few decades. Several machine learning methods have been used for protein secondary structure prediction including neural networks (NN), support vector machines (SVM), hidden markov models (HMM) and cascading models. A common underlying approach for all the secondary structure prediction methods is to extract statistical properties of amino acid distribution, physico-chemical properties of the residues of the protein sequences and then build models for classification. Initially Chou-Fasman and others [26, 64, 36] utilized neural networks and by using single amino acid sequence features

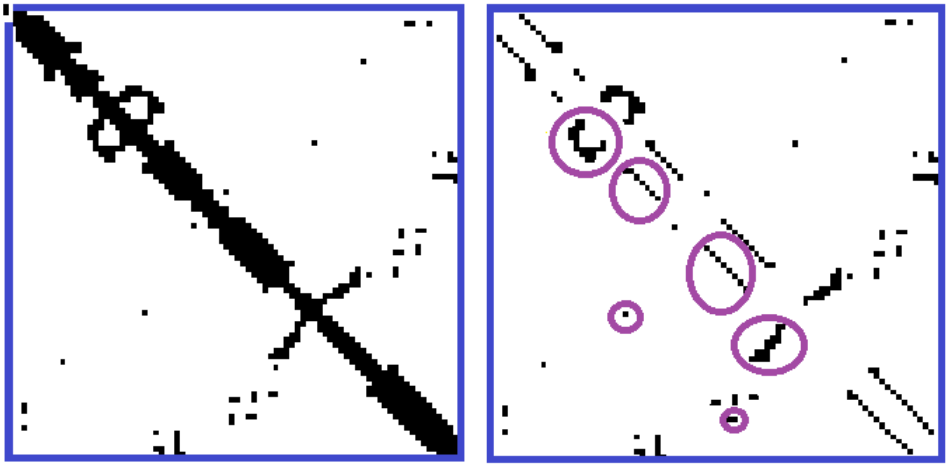


Figure 3.1: Diagonal and off-diagonal regions

achieved an accuracy of 50-63%. Jones et al. [45] consider multiple sequence features from the evolutionary information in the form of Position Specific Scoring Matrices(PSSM) generated by PSI-BLAST using PSIPRED algorithm to achieve an accuracy of 70%. Rost et al. [63] consider the Profile network from HeiDelberg (PHD) and by using a two-layer neural network with evolutionary information increased the accuracy by 1%. Karypis et al. [47] used cascaded Support Vector Machine (SVM) based predictor using PSI-BLAST profiles and proposed YASSPP algorithm. To summarize, average accuracy of secondary strucutre prediction has been in the range 71-80% so far.

In this chapter, we propose to extract features from both the diagonal region as well as the off-diagonal region of the contact map. Instead of using the standard clustering algorithms from literature, as contact maps are very sparse matrices, we propose to use heuristics on the diagonal and a naive algorithm to extract rectangular/ non-rectangular regions of connected pieces of contacts from the off-diagonal region.

3.3 Feature extraction along the diagonal region

A contact map can be divided into two regions: Diagonal region and the region obtained by masking the diagonal region referred to as the off-diagonal region in Figure 3.1.

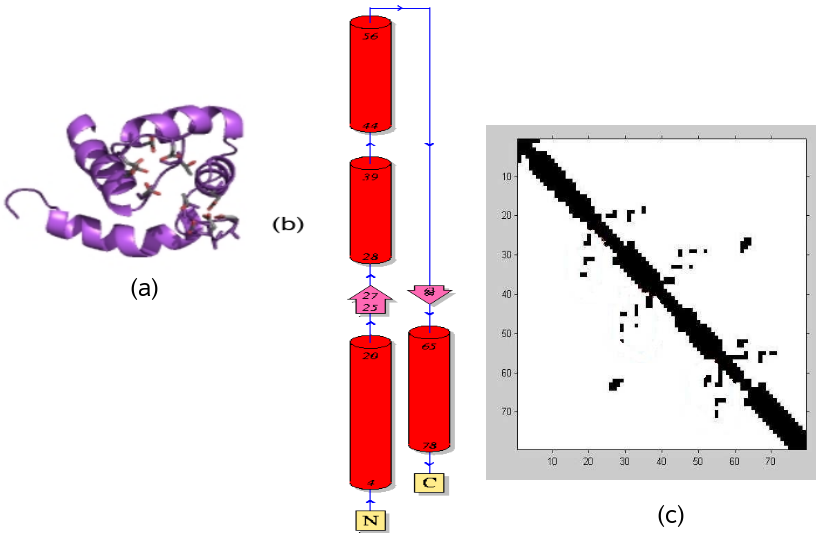


Figure 3.2: Protein 1SW8 of EF-Hand like Fold: (a) 3D structure, (b) topological cartoon, and (c) contact map of the PCN, where the black dots indicate interaction between the corresponding amino acid residues

3.3.1 Secondary structure prediction

Many researchers including Zaki et al. [97] and Hu et al. [42] emphasize that thick bands along the diagonal denote helices, and those that are away from the diagonal correspond to beta sheets. But actual extraction of these SSE from the contact maps has not been reported in the literature.

The 3D structure and the corresponding contact map of protein 1SW8 of EF-hand fold are shown in Figure 3.2. The thick bands along the diagonal corresponding to the four helical regions of the protein are clearly seen along the diagonal. The secondary structures are visible along the diagonal in the contact map and also the interactions between secondary structures are seen to be nicely embedded within the contact map.

3.3.2 Extract_SSP Algorithm

The idea underlying prediction of specific secondary structures of helix, beta and coils/turns is to extract, bands of width W and length l along the diagonal in the upper/lower triangular matrix of the contact map with parameters tuned to different secondary structure elements.

3.3.2.1 Fixing parameters

Typically since one turn of the helical structure is made up of 3.6 residues, the minimum predicted length for an α-helix should be three or four residues. Consecutive $C\alpha$ atoms

Algorithm 1 Extract_SSP(A)

Input: Contact Map A[$r \times c$]
Output: Secondary structure positions of A
Input parameters: row_width parameters: $a, b, 0 < b < a$
Variables: Helix_Length, Beta_Length, Coil_ Length, Number of helices: NH, Number of betas: NB, Number of turns/coils:NC all initialized to zero.

1: **for** $i \leftarrow 0$ to $r - 1$ **do**
2: $row\_width = 0$;
3: **for** $j \leftarrow 0$ to $c - 1$ **do**
4: **if** $(A[i][j] == 1)$ **then**
5: row_width++;
6: $A[i][j] == 0$;
7: **end if**
8: **end for**
9: **while** $(row\_width \geq a)$ **do**
10: Helix_Length++;
11: break; (go to next row)
12: **end while**
13: **if** $(row\_width < a)$ && $(Helix\_Length \geq 3)$ **then**
14: Print Helix Found;
15: NH++;
16: Reset Helix_Length=0;
17: **end if**
18: **while** $(b \leq row\_width < a)$ **do**
19: Beta_Length++;
20: break; (go to next row)
21: **end while**
22: **if** $(row\_width < b)$ && $(Beta\_Length \geq 3)$ **then**
23: Print Beta Found;
24: NB++;
25: Reset Beta_Length=0;
26: **end if**
27: **while** $(0 < row\_width < b)$ **do**
28: Coil_Length++;
29: break; (go to next row)
30: **end while**
31: **if** $(Coil\_Length > 0)$ **then**
32: Print Coil Found;
33: NC++;
34: Reset Coil_Length=0;
35: **end if**
36: **end for**

Protein	Original helices		Predicted helices					
			W=3		W=4		W=5	
1SW8	4	20	1	60	4	18	4	9
	28	39	64	77	28	37	10	18
	44	56			44	53	28	35
	65	79			64	76	48	52
							64	75

Table 3.1: 1SW8 protein has 4 helices, with original locations given in columns 2 and 3. Predicted helices for widths 3, 4, 5 show that only two helices are predicted for width = 3.

are farthest apart since, in a β-strand relatively few residues cross the protein core with a strand. Therefore, the number of residues in a β-strand is usually limited to two or three amino-acids [74, 97].

We conducted an initial scan of the data set for fixing the parameters on 10% of the data set, by fixing minimum helix length as 3 and varying helix width between the values 3, 4 and 5, potential helix regions have been extracted from the contact maps. For $W = 3$ we obtained many false positives with residues annotated as H, and for $W = 5$ in the overall protein, lesser number of helices were obtained. A sample result can be seen in Table 3.1. Hence we decided to fix the helix-width parameter as 4. Similarly, width of beta is set to 3 and minimum beta length as 3 using inputs from the literature [74, 97]

We propose *Extract_SSP* algorithm, in which we set the parameters of row_width a as 4 and minimum helix_length as 3. The beta strand prediction is also carried out by setting row_width b to be 3 and minimum beta_length as 3. All the remaining contacts are labeled as belonging to coil/turn.

In order to validate the algorithm we run *Extract_SSP(A)* on bench mark data set used in secondary structure prediction literature and compare the results with those obtained by some of the latest algorithms in the literature.

3.4 Experimentation on bench mark data set

We consider the gold standard data set RS126 of Rost [63], given in Table 3.2, which has been designed for the secondary structure prediction. This protein data set contains proteins that maintain pair-wise sequence similarity of less than 25%,

We use the standard evaluation measures like Q_3 and SOV_{99} (segment overlap) [98] for performance evaluation, the details of which are given below.

Table 3.2: Secondary structure prediction data set [63]

Class	Number of Proteins
All-Alpha	21
All-Beta	44
Alpha+ Beta	15
Alpha/ Beta	28
Small Proteins	13

3.4.1 Evaluation measures

Two kinds of performance measures have been frequently used in protein secondary structure prediction, viz Q_3 or accuracy (3 for the three types of secondary structures) and SOV Segment OVerlap based measure [98, 63]. Q_3 is a residue based measure which calculates the overall percentage of correctly classified residues for all the three structures, and is computed as follows:

$$Q_3 = \frac{H_{pre} + E_{pre} + C_{pre}}{N_{total}} \tag{3.1}$$

where N_{total} is the total number of predicted residues, H_{pre} is the number of correctly predicted residues for helix, E_{pre} for sheet, and C_{pre} for coil.

SOV measures the average length of helix-beta-coil segment overlap between predicted and actual sequences as shown in equation below. SOV differs from Q_3 during prediction since SOV penalizes wrong predictions, e.g., a single helix predicted as a multiply split helix gets lesser value with SOV.

$$SOV(i) = 100 \times \left[\frac{1}{N_i} \sum_{s(i)} \frac{minov_i\left(s_1, s_2\right) + \delta_i\left(s_1, s_2\right)}{maxov_i\left(s_1, s_2\right)} \times len\left(s_1\right) \right] \tag{3.2}$$

where s_1 and s_2 denote segments of secondary structure $i \in$ (H, E or C) N_i is a normalization value, $minov_i(s_1, s_2)$ is the length of actual overlap of structure i between s_1 and s_2; $maxov_i(s_1, s_2)$ is the length of total extent of i for s_1 and s_2 and $\delta(s_1, s_2)$ can be represented as formula shown below.

$$\delta\left(s_1, s_2\right) = min\left\{ maxov_i\left(s_1, s_2\right) - minov_i\left(s_1, s_2\right), minov_i\left(s_1, s_2\right), int\frac{len(s_1)}{2}, int\frac{len(s_2)}{2} \right\} \tag{3.3}$$

Let us consider an example of

Observed sequence : $\overbrace{HHH}^{\alpha_O^1} CCCCCC \overbrace{HHHHHH}^{\alpha_O^2}$

Predicted sequence: CCCCC$\overbrace{HHHHH}^{\alpha_P^1} CCC \overbrace{HH}^{\alpha_P^2}$

Accuracy calculation:
H_{pre} = 3 , C_{pre} = 2 , Total original residues =15
$Q_3 = \frac{5}{15}$
Accuracy Q_3 = 33.3%

Segment OVerlap measure calculation: In the observed sequence the first helix α_O^1 does not produce any overlapping pair, the second helix ($len(s_1) = 6$) produces two of them:(α_O^2, α_P^1) and (α_O^2, α_P^2). The value of $SOV(H)$ is calculated as follows:
for (α_O^2, α_P^1): $\delta(s_1, s_2) = 1$, $minov = 1$, $maxov = 10$,
for (α_O^2, α_P^2): $\delta(s_1, s_2) = 2$, $minov = 2$, $maxov = 6$ and
$N_i = \frac{1}{15} = 0.066$.
$SOV = 100 \times (0.066 * 0.86 * 6)$
Segment OVerlap SOV= 34.056%

3.4.2 Results and discussion

Extract_SSP is run on the data set RS126. The class wise secondary structure prediction test set results are given in Table 3.3. It can be observed that highest performance is achieved for alpha+beta class with Q_3 as 88% and SOV as 89%. It can be seen that helix prediction is seen to be much higher in all the classes with respect to both measures of Q_3 and SOV, whereas turn/coil prediction seems to be working reasonably only for All-Alpha class. The features do not seem work well at all for turn/coil prediction.

We compare the results obtained by our algorithm to the existing algorithms in which the results have been reported for the data set RS126. We can see from Table 3.4 that our algorithm is performing on par with the other algorithms with prediction for helix H_{SOV} being much higher than the results reported and Q_3 and SOV being on par with the results. The algorithm is seen to perform very poorly for turn/coil prediction. In protein structures one finds very long coils which are referred to as loops. A loop connects two secondary structural elements. We observe that many turns/coils have been misclassified as beta in our test.

Table 3.3: Performance of secondary structure prediction algorithm on each structural class of proteins

Class	Measure	Over All	H	E	C
All-Alpha	Q_3	83	88	-	77
	SOV	85	94	-	82
All-Beta	Q_3	67	74	71	35
	SOV	68	80	79	26
Alpha+Beta	Q_3	88	91	74	31
	SOV	89	93	75	32
Alpha/Beta	Q_3	67	89	67	34
	SOV	62	96	58	38
Small Proteins	Q_3	76	87	74	46
	SOV	76	91	73	46

Table 3.4: Comparison between proposed and other methods

Methods	Q_3	SOV_{94}	H_{sov}	E_{sov}	C_{sov}
PHD [63]	71	74	72	66	72
SVMfreq [43]	71	75	73	58	73
SVMpsi [48]	76	**80**	**80**	**72**	**73**
YASSPP [90]	**77**	71	71	63	55
Proposed	**76**	**79**	**91**	**73**	**56**

3.5 Results for predicted contact maps

In our thesis, instead of deriving features from amino acid sequences, we assumed that the contact maps can be predicted from the amino acid sequences. Hence we propose a new methodology by deriving features only from protein contact maps in order to predict secondary structures. So far, we considered contact maps that have been derived from the 3D structures available at PDB [5]. For the work to be meaningful, we have to repeat the prediction by testing on the contact maps that have been predicted from the primary sequence. It was not easy to obtain predicted contact maps from researchers. Only recently, we found a server namely Distill, made available by Bau et al. at University College Dublin which we have used to construct a small data set of predicted contact maps.

We choose 15% of proteins randomly from each of the five classes to construct a data set of size 18 proteins. As it was a time consuming exercise to obtain predict contact maps, we limited our experimentation to 18 proteins from the RS126 [63]. Each of these protein sequences is submitted to *Distill* [3] software to obtain the contact matrices. These contact maps are then submitted to our *Extract_SSP* algorithm for secondary structure prediction.

These results are tabulated in Table 3.5.

The secondary structure prediction test set results for proteins of each structural class are given in Table 3.5. It can be observed that satisfactory result of an overall accuracy of 76% is obtained with highest performance achieved for All-Alpha and Alpha+Beta classes with Q_3 as 83% and SOV as 89%. It can be seen that helix prediction is seen to be much higher in all the classes with respect to both measures of Q_3 and SOV,

Table 3.5: Performance of Secondary structure prediction algorithm on Predicted contact maps [63]

Class	Measure	Over All	H	E	C
All-Alpha	Q_3	83	88	-	77
	SOV	89	94	-	82
All-Beta	Q_3	67	74	71	35
	SOV	68	80	79	26
Alpha+Beta	Q_3	83	91	74	31
	SOV	89	93	75	32
Alpha/Beta	Q_3	67	89	67	34
	SOV	62	96	58	38
Small Proteins	Q_3	76	87	74	46
	SOV	76	91	73	46

Hence, secondary structure elements can be extracted from the diagonal region whose statistics are intended to be used as features.

3.6 Feature extraction along off-diagonal region

The clusters of 1's seen away from the diagonal represent interactions between different secondary structures due to the folding of the protein chain. Hu et al. [42] mine contact maps and show that the off-diagonal interactions can be annotated as being $\alpha-\alpha, \alpha-\beta$ and $\beta-\beta$ interactions among the secondary structure elements (SSE). Shi et al. [71] proposed region based approach in the literature and perform the structural class classification. We consider the lower triangle region of off-diagonal interactions as region of interest (ROI) to show that there are significant differences among these regions for proteins of different folds. Suppose we divide the triangle into four equal triangles namely Left(L), Right(R), Top(T) and Middle(M). Figure 3.3 shows the contact maps of 1SW8 protein on the left and Cytochrome -C protein 451C on right. Clearly the T and R triangles in the contact map of Cytochrome -C fold show distinct looking patterns. As an initial experiment, we tabulate the density of patterns for both these folds in Tables 3.6 and 3.7. There is a

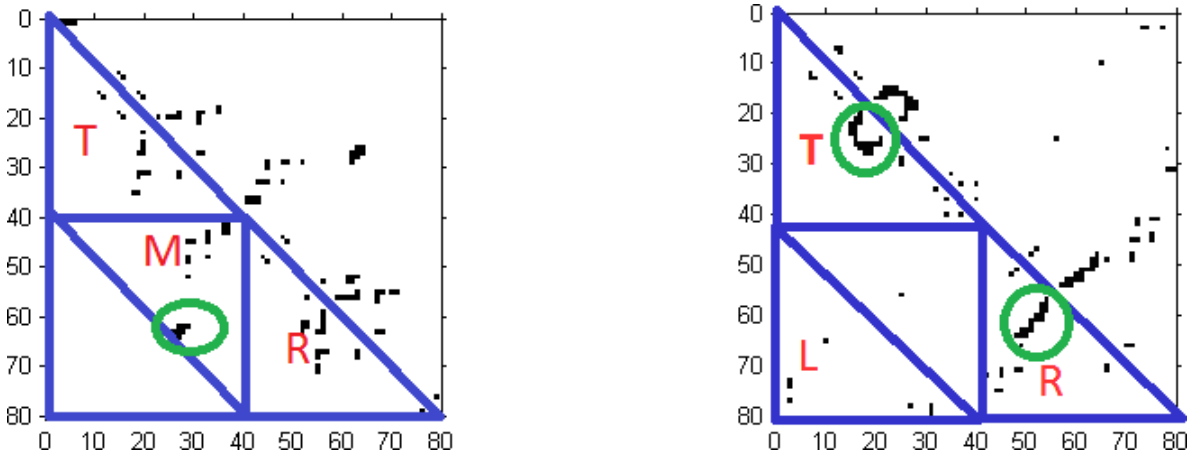


Figure 3.3: The contact maps of (a) protein 1SW8 of EF-hand fold showing majority of interactions in the triangles T, M, R and (b) protein 451C of Cytochrome -C fold showing off-diagonal patterns in the triangles T, L, R. The circled patterns indicate interactions between a pair of helices significant to the respective fold.

striking difference seen between the two folds even at this superficial level with T, M and R dominating in EF-hand fold where as T, L and R dominating for Cytochrome -C fold. There is a need for deeper investigation into the densities and shapes of the patterns in the off-diagonal region.

Table 3.6: The density of interactions in the off-diagonal region (=T, M and R sub-triangles) in proteins of EF-hand like fold.

PID	TOP	LEFT	MIDDLE	RIGHT
1S6j	32	8	16	31
1Clm	91	0	4	94
1Sw8	33	1	23	51
1R2u	33	6	21	39
1Oqp	38	7	17	40
1Mxe	89	3	5	94
1Cdm	85	1	6	99
1N0y	42	2	14	52
1Ggz	91	0	4	103
1R6p	33	6	23	38

We extract continuous regions of contacts from the off-diagonal region of the contact maps using a simple and computationally inexpensive algorithm called *ExtractPatterns*. We believe that these regions hold significant information with respect to the structural class and fold information of the protein.

Table 3.7: The density of interactions in the off-diagonal region (=T, L and R sub-triangles) in proteins of Cytochrome -C fold.

PID	TOP	LEFT	MIDDLE	RIGHT
1GDV	43	20	4	53
1S91	53	22	8	55
1CCR	48	39	24	53
1CO6	52	37	26	47
1COT	66	35	31	63
451C	50	14	3	51
1CC5	43	35	8	46
1CTJ	46	26	4	53
1CYJ	51	23	4	54
1LS9	53	22	8	55

3.6.1 Pattern extraction algorithm

Once the secondary structure prediction is completed, the diagonal of the contact map is masked by placing zeros along a band of width 4.

Algorithm 2 ExtractPatterns(A)

Input: Two dimensional masked contact matrix $A[i][j] : 0 \leq i \leq r-1, 0 \leq j \leq c-1$
Output: Pattern Database DB, Number-of-patterns in A: $np(A)$, density(Pattern)
Input parameters: d: Minimum density

```
1:  for i ← 0 to r − 1 do
2:      for j ← 0 to c − 1 do
3:          if (A[i][j] == 1) AND (i ≤ j) then
4:              Set nc(A) ← 0 /* Pattern Counting*/
5:              Pattern(i, j);
6:              if density(Pattern(i, j)) ≥ d then
7:                  𝔻𝔹 = 𝔻𝔹∪ PatternArray[i,j]
8:                  np++;
9:              end if
10:         end if
11:     end for
12: end for
```

The *ExtractPatterns* algorithm finds patterns of 1's in the contact map as follows: If bit position is '1' then mask the corresponding bit and read the neighboring pixel. This process is repeated for the Moore-neighborhood of the bit, until a whole continuous pattern is extracted from Figure 3.4. Clearly, this algorithm does not split any pattern cluster.

Algorithm 3 Pattern (x, y)

1: $P[x][y] \leftarrow 1$
2: $d \leftarrow d + 1$;
3: $A[x][y] \leftarrow 0$ /* (x,y) is read and hence disabled.*/
4: **for** each((u,v)=non-zero moore-neighbor(x,y)) **do**
5: $Pattern(u, v)$;
6: **end for**

Figure 3.4: Extract whole continuous pattern

Further the number of 1's in a pattern denotes density of pattern, the algorithm ensures that only patterns with a minimum density d are stored.

A few examples of patterns extracted from the protein sequences of the bench-mark data set are shown here. We can see that patterns extracted are rectangular/non-rectangular continuous regions of 1's.

$$Pattern1 = \begin{array}{cccccccccc} & & 1 & 1 & 1 & & & 1 & & \\ & & 1 & 1 & 1 & & & 1 & 1 & \\ & 1 & 1 & 1 & & & 1 & 1 & & \\ 1 & 1 & 1 & 1 & & & & 1 & & \\ 1 & 1 & 1 & 1 & 1 & & & 1 & 1 & \\ & & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ & & 1 & 1 & 1 & & & 1 & 1 & 1 \\ & & 1 & 1 & 1 & & & & 1 & 1 \\ & & 1 & 1 & 1 & 1 & & & & 1 \\ & & 1 & 1 & 1 & 1 & 1 & & 1 & \end{array}$$

$$Pattern2 = \begin{matrix} 1 & 1 & \\ 1 & 1 & \\ & 1 & 1 \\ & 1 & 1 \\ & & 1 \end{matrix}$$

$$Pattern3 = \begin{matrix} 1 & & \\ 1 & & \\ & 1 & \\ 1 & 1 & \\ 1 & 1 & 1 \end{matrix}$$

$$Pattern4 = \begin{matrix} 1 & 1 \\ 1 & 1 \end{matrix}$$

3.6.1.1 Pattern features

Three features are extracted from each protein, namely the number of patterns, minimum pattern density and maximum pattern density using *ExtractPatterns* algorithm.

3.6.1.2 Directional features

The figure on the left in Figure 3.5 shows the Moore-neighborhood of the eight directional bit positions. The co-variance of each pattern in the contact map is computed in order to predict the direction of the pattern as to whether the pattern is parallel or orthogonal to the diagonal. If the co-variance value is negative, the pattern is labeled as anti-parallel sheet. If the co-variance value is positive, the pattern is identified as parallel sheet. If it is zero the pattern can be either parallel or anti-parallel. Figure 3.5 shows the contact map of the protein 2IGD. The blue color patterns represent the anti-parallel sheets, red color pattern represents the parallel sheet and the green color dots represent the helix information along the diagonal.

3.7 Conclusion

This work validates the hypothesis that contact maps contain useful information that can be utilized to understand the problem of protein fold prediction. Secondary structure elements of helices and beta strands have been successfully extracted using the pattern information

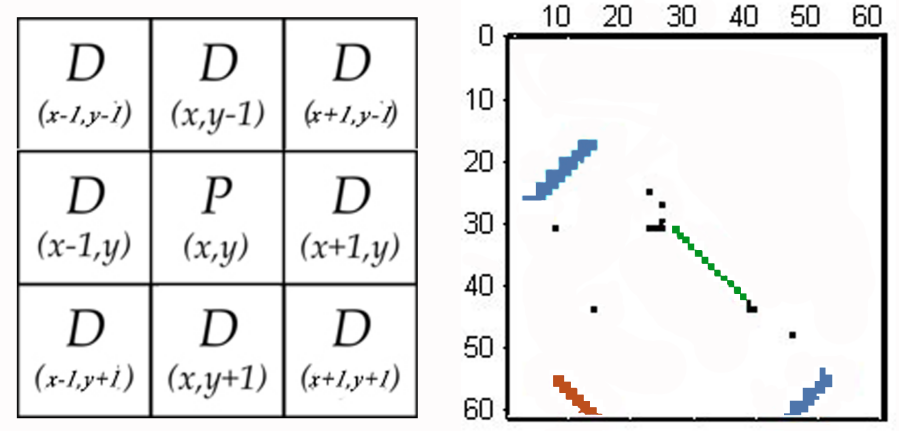


Figure 3.5: Eight directional bit positions and protein 2IGD diagonal masked contact map

in the contact map. On the other hand, coils could not be extracted well. Hence we do not use the statistics of coil/turn in our work. This issue needs to be looked into further.

TSM method shows that the off-diagonal region contains significant information to be exploited for further analysis. We have extracted (non)rectangular patterns from the off-diagonal region using *ExtractPatterns* algorithm. Several useful features relating to both secondary structures viz. number of helices, minimum helix length, maximum helix length, number of beta sheets, minimum beta sheet and maximum beta sheet; as well as pattern features like number of patterns, minimum and maximum density as well directional features have been extracted. These are going to be used as features for the fold prediction problem in the subsequent chapters.

Chapter 4

Multi-class Classification: Protein Fold Recognition

4.1 Introduction

Multi-class classification problem with imbalanced data is a challenging problem. This chapter addresses the problem of protein fold recognition which is a multiclass classification problem having unbalanced classes. Imbalanced data distribution involves a few of the classes of data having very few training samples compared to other classes. Support Vector Machines are believed to be less prone to the class imbalance problem than other classification learning algorithms, since boundaries between classes are calculated with respect to only a few support vectors and the class sizes may not affect the class boundary too much [44]. Decision tree classifiers have also been explored for imbalanced problems [23] in which boosting technique has been suggested. The basic idea of boosting is to repeatedly apply a weak learner to modified versions of the training data, and then take a weighted sum of the results of weak classifiers.

Most of the state-of-the-art research for protein fold recognition problem has focused on developing novel features based on amino acid composition [70] and other sequence features and use classifiers such as Support Vector Machines [69, 25], Decision trees [50] and Neural Networks [29]. We propose a novel way of solving this classification problem. Using the premise that contact maps can be predicted from the primary sequence of a protein, we carried out feature extraction from the contact maps in the last chapter. We use these pattern-features extracted from the diagonal region and the off-diagonal region of the contact map instead of using any chemical knowledge of amino acid residues, and

use boosting algorithms which have not been explored much for this problem in order to improve performance for protein fold recognition problem.

4.2 Literature

4.2.1 General approaches

One of the standard bench-mark data sets for the problem of protein fold recognition is constructed by Ding et al. [29]. They formulate the problem as a 27-way classification problem and use several multi-class methods like one-vs-all, all-vs-all classifiers. Ding et al. [29] extract six types of feature sets from protein sequences. They show that using these multiple feature sets and applying majority voting scheme can achieve a prediction accuracy of about 51.1%. Shamim et al. [69] consider both sequence and secondary structural features like frequencies of amino acids and amino acid pairs, secondary structural state and solvent accessibility state, to compose a feature vector of size 100 and obtain on accuracy of 60.5% on average on the data set of Ding et al. and 71% on an extended data set. Clearly data set of Ding et al. issues a challenge for classification algorithms. Improving the prediction accuracy for this problem has proved quite hard. Shamim et al. considered protein sequence features involving amino acid composition(C), hydrophobicity(H), polarity(P), predicted secondary structure(S), Vander Waals(V) volume and polarizability(Z). Further, they consider composition of dipeptide features which amount to 400 features. Combinations of these features go up from 20 to 4,900. Results are reported for these different sets of features. Best results are obtained for a feature set containing 100 features. Krisnaraj et al.[50] consider the feature sets of Ding et al. which vary in sizes between 41 to 125.

In comparison, we consider only 11 features involving count, length and densities of 2D-patterns that are extracted from the contact map.

4.2.2 Approaches with Boosting

Boosting is a very popular technique for improving the accuracy of classification problems with imbalanced data. Freund and Schapire [65] proposed a prize-winning boosting algorithm called AdaBoost. Another popular boosting algorithm is that of Chawla et al. [21] who introduced a novel oversampling technique named Synthetic Minority Over-sampling Technique (SMOTE) which is specifically designed for learning from imbalanced data sets. Literature reports that AdaBoost and SMOTE algorithms have been successfully applied to most popular classifiers [30, 35, 66, 67]. Many enhancements have been carried out which

include AdaCost [31], CSB1 and CSB2 [77], and RareBoost [46]; and SMOTEBoost [40], and DataBoost-IM [40]. AdaBoost and SMOTE are both stated to be capable of bias and variance reduction [22].

Krishnaraj [50] et al. are the first researchers to consider applying boosting techniques for improving the accuracy of protein fold recognition problem. They use the decision tree classifier of J48 and using boosting algorithm of AdaBoost, obtain on accuracy of 59%, and 60.3% using LogiBoost algorithm. Dehzangi et al. [27] use the bagging technique to increase accuracy and achieve 64.5% using an ensemble classifier with majority voting. Very recently Sharma et al. [70] throw attention back on amino acid physio-chemical attributes and use successive features selection method, but do not show any increase in the overall accuracy.

We consider novel features extracted from the contact maps, and then using C4.5 decision tree classifier with SMOTE boosting algorithm, a combination strongly suggested by Chawla et al. [21], perform protein fold classification. All the experimentation is carried out on the bench-mark data set of Ding et al. for facility of comparison, the details of which are given below.

Useful features are to be computed for these proteins. In the literature, amino acid composition, physico-chemical properties and secondary structural properties have been considered as features for classification. We derive secondary structure related features from the diagonal and the off-diagonal region of the contact maps.

4.3 Feature set

In the previous chapter 3.3.1, many features relating to secondary structure information have been derived along the diagonal. Further, off-diagonal features relating to patterns of contacts have also been extracted. The *ExtractPatterns* algorithm is run on the protein fold data set given in Table 4.1 to extract pattern features with different density thresholds varied between 2 to 5. With lower thresholds, too many patterns are obtained and with $d = 5$, too few hence we set the minimum density threshold as $d = 4$. A pattern data base (DB) containing 512 patterns is obtained with $d = 4$.

We constitute the feature set using this information in order to carry out fold recognition.

Table 4.1: Data set of Ding et al. [29] representing the four structural classes of proteins

Structural Class	Number of Folds	Training set	Test set	Total Proteins
All-Alpha	6	68	48	116
All-Beta	9	109	91	200
Alpha/Beta	9	115	109	224
Alpha+Beta	3	38	62	100
Total	**27**	**330**	**310**	**640**

- **Diagonal features(6)**:
  Secondary structural features from the diagonal region of the contact map: number of helices, minimum and maximum helix length, number of betas, minimum and maximum beta length .

- **Off-diagonal features(5)**:
  Number of patterns, minimum pattern density, maximum pattern density, number of parallel and anti-parallel sheets .

- **Feature set : Diagonal + Off-diagonal(11)** Number of helices, minimum and maximum helix length, number of betas, minimum and maximum beta length, number of patterns, minimum pattern density, maximum pattern density, number of parallel and number of anti-parallel sheets .

4.4 Classification and results

Ding et al. discuss the several multi-class methods like one-against-one, one-against-others and all-versus-all methods for classification [29]. These methods are not enough to improve prediction accuracy because of inadequacy of instances among classes in data set. We apply SMOTE based technique as a boosting technique to address the imbalance in the data. The SMOTE algorithm re-balances the inadequate classes. Firstly, the minority class is over sampled by taking samples from the minority class and then introducing synthetic examples along the nearest neighbors of the minority class. Secondly, the majority class is under sampled by randomly removing samples from majority class until the minority class grows unto a specified percentage of the majority class. Thus a combination of under-sampling and over-sampling leads to the initial bias of the learner towards the majority (negative) class being reversed in favor of the minority (positive) class.

SMOTE is applied to the training data set containing 330 instances and the data after boosting increased to 617 instances. Now J48 classifier is trained on this data set with all-versus-all classification using 10-fold cross-validation. That is, the data set is divided randomly into 10 equal parts out of which the model is trained on 9 parts and tested on the remaining part. This experiment is repeated 10 times so that the model learns to generalize and over-fitting is avoided. The test set of Ding et al. is used for testing the classifier that was trained on the boosted data and the results are discussed in the next section.

4.4.1 Evaluation measures

The evaluation measures are the standard measures of Precision, Recall and F-measure which are based on the confusion matrix. TP and TN denote the number of positive and negative instances which are classified correctly; FN and FP represent the number of mis-classified instances of positive and negative classes respectively. Recall gives the proportion of positives out of the total instances predicted as positive and is calculated as

$$Recall(R) = \frac{TP}{TP + FN}$$

Precision is the percentage of positive predictions that are correct. It is calculated as

$$Precision(P) = \frac{TP}{TP + FP}$$

Finally F-measure is calculated by using the formula

$$F - measure = \frac{2 * R * P}{R + P}$$

Two special measures called Geometric Mean(GM) and Area under the Curve(AUC) have been proposed by Kubat et al. [51] in the context of imbalanced data. GM(1) defined as $\sqrt{TP * P}$ and GM(2) as $\sqrt{TP * TN}$ turn out to be useful to evaluate a classifier in the case of unbalanced classification problems.

Area under ROC curve (AUC) is often used as a measure of quality of the classification models. A random classifier has an area under curve as 0.5, while AUC for a perfect classifier is equal to 1. In practice, most of the classification models have an AUC between 0.5 and 1. It measures the discriminating ability of a binary classification model. The AUC measure is useful for data sets with unbalanced target distribution in which one target class dominates the other.

4.4.2 Results and discussion

The classification results with Diagonal+Off-diagonal features are presented in this section.

The classification results are shown in Table 4.2. The results show that highest recall of 75% and GM of 68.2% are obtained for All alpha class and lowest recall of 54% for Alpha/Beta. As seen in Sec 3.3.1, since beta sheets have not been detected well on par with alpha-helices, and this may be limiting the prediction accuracy results in this case. Though AUC shows good values, the measures of Recall Geometric Mean show the true picture regarding the proportion of true positives which needs to be improved.

Table 4.2: Classification results using Diagonal + Off-diagonal features

Class	Precision %	Recall %	F-Measure	AUC%	Geometric Mean(GM)
All-Alpha	85	74.8	79.33	89.4	68.2
All-Beta	76	69.4	71.16	78.5	60.1
Alpha/Beta	76	53.5	62.58	78.8	46.2
Alpha+Beta	92.33	63.3	74.15	82.6	60.4
Average	**82.33**	**65.25**	**71.80**	**82.32**	**58.7**

Table 4.3, carries a comparison of performance with existing literature among which the method of Krishnaraj [50] et al. alone used boosting technique. We compare the accuracy (Recall) values of all the algorithms to get an idea of how much boosting technique helps in improving the performance of the classifier. The other methods in literature report performance values for Recall and Krishnaraj et al. do not give class-wise results and hence only the results that are available are given in the comparison table.

Table 4.3: Classification results of Recall obtained on 4-major structural classes for the proposed method as well as results from literature

Class	Shamim [69]	Ding [29]	Krishnaraj [50]	Proposed Method
No. of features	100	62	104	**11**
Average Recall	60.5	51.1	60.3	**65.25**

The boosting technique along with the novel features boosts up the accuracy of the classifier to 65.25% using only 11 features. Details of results within each class are given in Table 4.4 giving the 27-way classification results for each protein fold.

Note that All-Alpha class shows highest recall of 87% for globin like fold and lowest recall of 64% for Cytochrome-C fold. All-Beta class shows a recall of above 80% in cupredoxins fold, viral coat and capsid proteins, and sh3-like barrel folds. The number of proteins in these classes is less so more synthetic samples must have been generated. So, true positive rate is high thus increasing the classification accuracy. Lowest recall is 20% in

Immunoglobin-like beta-sandwich fold in which both true positive rate and false positive rates are very low. Alpha/Beta class shows the highest recall of 66% for Thioredoxin-like and 61% for FAD binding motif folds. Alpha+Beta class shows a highest recall of 75% for Ferredoxin-like fold and a lowest recall of 42% in small inhibitors fold.

Table 4.4: 27-way classification results

			All-Alpha Class	
Fold	Fold Name	Precision	Recall	F-Measure
1	Globin-like	87	87	87
3	Cytochrome-C	86	64	73.38
4	DNA/RNA binding 3-helical bundle	88	85	84.47
7	Four helical up-and-down bundle	77	66	71.07
9	Four helical cytokines	80	75	7741
11	EF-Hand	92	72	80.70
			All-Beta Class	
20	Immunoglobulin-like β-sandwich	23	20	22
23	Cupredoxins	80	83	81
26	Viral coat and capsid proteins	84	81	83
30	ConA-like lectins/glucanases	88	70	77.97
31	SH3-like barrel	91	87	89
32	OB-fold	86	86	86
33	Trefoil	86	57	68.5
35	Trypsin-like serine proteases	57	66	61
39	Lipocalins	89	75	72
			Alpha/Beta Class	
46	(TIM)-barrel	20	19	19
47	FAD (also NAD)-binding motif	88	61	72.05
48	Flavodoxin-like	79	60	68.20
51	NAD(P)-binding Rossmann-fold	89	59	70.95
54	P-loop containing nucleotide	90	50	64
57	Thioredoxin-like	76	66	71
58	Ribonuclease H-like motif	88	62	72.74
62	Hydrolases	88	60	71.35
69	Periplasmic binding protein-like	69	45	54
			Alpha+Beta Class	
72	β-grasp	90	73	80.61
87	Ferredoxin-like	97	75	84.59
110	Small inhibitors, toxins, lectins	90	42	57.27
	Average	**82.33**	**65.25**	**71.80**

When we analyzed the low recall results, we make a general observation among all these classes. The classifier seems to be performing better with respect to the classes having more synthetic samples, since, the synthetically generated samples may be similar to the other data in the sample. On the other hand, the folds 20, 32 and 46 each have more number of original sequences and hence cannot take advantage of boosting and hence report less accuracy.

Now we test the classifier on the data set of predicted contact maps.

4.5 Results for predicted contact maps

We consider the test set data set of All-Alpha class sequences from Ding et al. [29]. This data set contains 48 sequences in All-Alpha class. We submitted these to Distill [3] software for obtaining the predicted contact matrices. Extract_SSP algorithms and ExtractPatterns have been applied to these predicted contact maps to compute the secondary structure and pattern features. By using J48 decision tree model that has been already trained, we test the performance on this set. The accuracy results are tabulated in Table 4.5. Highest accuracy of 90% is seen for EF-Hand like fold. Globin like and Cytochrome -C folds achieve the same accuracy of 87%. Four helical up and down bundle got less accuracy because false positives are high. The results for predicted contact maps do not deviate from

Table 4.5: Test performance on predicted contact maps from All-Alpha class

Fold index	Fold name	Precision	Recall	F-Measure
1	Globin-like	87	87	87
3	Cytochrome C	87	87	87
4	DNA-binding 3-helical bundle	66	81	73
7	4-helical up-and-down bundle	77	58	71
9	4-helical cytokines	60	65	67
11	EF-hand	79	90	84
Average		**76**	**78**	**78**

the overall results seen in Table 4.5 which validates our whole approach and ideas. In fact in most cases, the test results on the predicted contact maps turn out be much better than the contact maps constructed directly from the structure.

4.6 Conclusion

In order to address the multi-class classification problem, we experimented with several techniques like one-against-one, one-against-others and all-versus-all methods given in the literature. We found that these methods are not adequate to improve prediction accuracy because of imbalanced data present among the classes. We decided to use boosting approaches to this problem. In the literature, we found that only one paper reported results with boosting for protein fold recognition problem. The literature on this problem also used a number of features derived from amino acid composition and secondary structure related features ranging between 100 to 4900 features. The best results reported by Shamim et al. used around 100 features in order to obtain an accuracy of 60.5% on the bench-mark data set of Ding et al. and obtain 71% on an extended data set using combination of features adding upto 4900 features [69].

We claim that contact map features proved to be useful for classification since with 62 amino acid features along with boosting, Krishnaraj et al. obtained 60.3% on average whereas our method achieves an increased accuracy of 65.25% for this challenging problem of protein fold recognition. We do not experiment with highly improved versions of boosting algorithms that are available now [22] which may improve the results even further. Our emphasis is more on features derived from contact maps rather than on using state-of-art classifiers. Obviously there is so much scope for improvement. We had experimented with only diagonal features and we found to our surprise that on Alpha/Beta class, the classifier gives an average accuracy of 92% which fell down to 54% when off-diagonal features were included. This issue certainly needs to be looked into carefully. We conduct a deeper investigation into the off-diagonal pattern features in the next chapter. in much lesser time when compared to the training of state-of-the-art models such as Support Vector Machines or Neural Networks.

Chapter 5

Association Rule Mining for Protein Fold Signatures

5.1 Introduction

The aim of this study is to extract 'signature' of a protein fold using only 2D-pattern features extracted from contact matrices. In the previous chapter, we built a model in a typical 'black-box' machine learning approach. In this work, we would like to open up the black-box and extract rules that are inherently guiding the fold identification.

Barah et al. were the first to indicate that contact map analysis could be used for fold recognition [14]. They demonstrate how the conserved contact patterns within proteins of a fold look similar visually and emphasize the hypothesis that proteins belonging to the same fold, have similar contact maps. Figure 5.1 shows two different proteins of Cytochrome C fold having distinct looking patterns which are similar in both the contact matrices. Can we generalize these ideas and extract rules involving conserved patterns specific to a particular fold? Clearly a closer study of contact maps may help in deriving rules that pertain to fold information.

Turcotte et al. [80] propose rules for fold prediction based on Inductive Logic Programming (ILP) using features pertaining to the protein 3D-structure. We follow a similar approach and intend to eventually derive rules that correspond to each fold of a protein structural class. Then these rules can be tested on empirically predicted contact maps [85, 99] to identify folds. These ideas lead to a new approach to protein fold classification using association rule mining of protein contact maps.

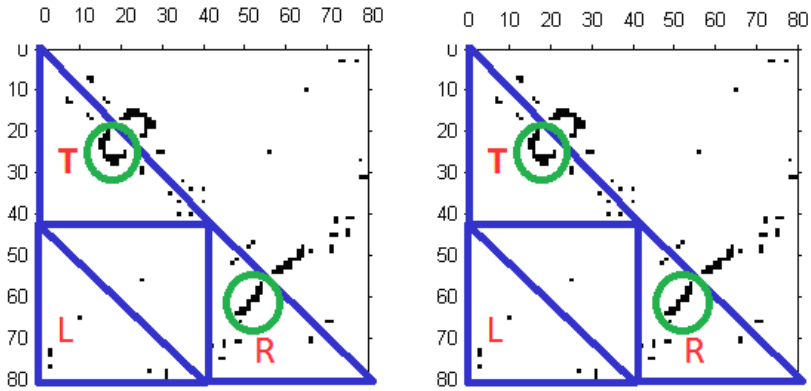


Figure 5.1: Cytochrome -C proteins of 351C and 451C

5.2 Literature

5.2.1 Graph similarity approaches

Graph similarity is a problem that has been addressed by many researchers. Graphs G_1 and G_2 can be assessed for similarity through graph isomorphism or sub-graph isomorphism approaches. Maximum common sub graphs can also be used for computing similarity. These formulations are not useful as most of these are NP-Hard problems. In this context, work on approximate graph matching based on vertex similarity assumes importance [49, 17, 95, 68]. Other popular approaches are eigen-vector based methods [72].

We propose that adjacency matrices corresponding to protein contact maps need not be treated with these time-consuming methods. These matrices are very sparse and their matrix representation shows small distinct looking clusters in the off-diagonal region as shown in chapter 3.6, which can be considered separately rather than taking the graph as a whole. Graph-theoretic approaches have been applied to protein graphs for addressing protein structure comparison problem. Node clusters of interest, for example corresponding to active sites of proteins, have been retrieved using clustering algorithms [86]. In this thesis instead of considering clusters of nodes as a query, we consider clusters of edges i.e contacts which we refer to as 2D-patterns for identifying protein folds. With this approach algorithms turn out to be linear as compared to the conventional approaches in which retrieval of nodes of interest becomes equivalent to sub-graph isomorphism problem which is NP-hard. It is surprising to see that the non-trivial 2D-patterns extracted capture the conserved interactions within a fold leading to fold recognition.

5.2.2 Sparse matrix approaches

Hu et al. [42] suggest mining of protein contact maps to extract the 2D sub matrices. Zaki et al. [97] evolve novel graph based methods for discovering protein folding pathway and indicate that it should be possible to formulate rules to identify secondary structures in contact maps [97]. Zaki [96] uses contact maps to discover an extensive set of dense patterns. They extract 2D sub-matrices of a fixed-size, say, w, from a contact matrix of size n. These patterns are clustered based on their similarity and clustering quality. They propose a fast hashing scheme for indexing and redundancy removal while storing the pattern in the data base [96].

Fraser et al. [33, 34] conducted a study to identify specific regions within contact maps. The authors choose contacts corresponding to alpha-alpha interactions in 171 proteins and demonstrate that these exhibit high similarity with the help of Jacquard and cosine metrics. They study $\alpha - \alpha$ interactions in depth and typify them further as corner, edge and central.

On the one hand, Zaki's scheme gives patterns of fixed size, and on the other, it is a time-consuming step of order $\frac{1}{2}((n-w) \times (n-w))$, and further, may not retrieve a structural pattern as a whole. We propose that incorporating redundancy removal in our *ExtractPatterns* algorithm, we can build a much smaller pattern database very efficiently and which contains patterns having atleast 33% dissimilarity.

5.3 Building pattern database

We construct a data base containing all the 'significantly different' patterns that occur in the off-diagonal region of the protein contact maps from the bench mark data set of Ding et al. Each protein is going to be represented as a feature vector using these patterns as features, with presence of a pattern denoted as 1 and absence by 0. Thus this step of pattern extraction is a crucial step to the entire process.

5.3.1 Data set

The data set created by Ding et al. [29] has been introduced in Table 4.1. The data set contains 27 folds with 640 proteins belonging to the four structural classes All-Alpha, All-Beta, Alpha/Beta and Alpha+Beta. These protein structures have been downloaded from the protein data bank [5] for which the contact matrices are computed.

Continuous 2D-patterns are extracted from the contact maps using *ExtractPatterns* algorithm as given in Section 3.6.1 which are further pruned by removing similar patterns.

5.3.2 Extract significant patterns

We refine the *ExtractPatterns* algorithm by not storing a pattern P' if the database already contains P of same size as P' and having hamming distance less than a threshold value t. This procedure is given in *Sig_Pattern* Algorithm.

5.3.3 Algorithm

Algorithm 4 $Sig\_Pattern$

for each protein $A \in Dataset$ **do**
  $C \leftarrow$ *ExtractPatterns*(A);
  /*Set C contains patterns P extracted from A*/
  Store a pattern P of C in $Pattern\_DB$
  /*Check for redundancy of the pattern*/
  for each pattern $P' \in Pattern\_DB$ **do**
    for each pattern P in C **do**
      if $\mid P \mid \neq \mid P' \mid$ **then**
        Store P in $Pattern\_DB$
      else
        if Ham-dist$(P', P) \geq t$ **then**
          Store P in $Pattern\_DB$/* Found significantly different pattern */
        end if
      end if
    end for
  end for
end for
RETURN $Pattern\_DB$

We run the $Sig\_Pattern$ algorithm on the entire Data set 4.1 with a choice of $t = 33\%$ to build the pattern database $Pattern\_DB$. In Table 5.1, we present a few large patterns extracted using the algorithm.

5.4 Significance of the extracted patterns: A few case studies

We study some of the conserved patterns obtained by the *Sig_Pattern* algorithm for their structural significance. We present details of the protein contact maps belonging to the major structural class of All-Alpha (given in Table 5.2).

Table 5.1: A few large patterns found in the off-diagonal region of contact maps of All-Alpha class

Name of the Fold	Pattern	Structural significance
Cytochrome -C	0000000000000111 0000000000011100 0000000000111000 0000000011110000 0000000011100000 0000001111000000 0000001110000000 0000011100000000 0000111000000000 0001111000000000 1111110000000000 0011000000000000	Beta-Beta-Interaction
DNA/RNA binding 3-helical bundle	0000000000000010 0000000011111111 0000000011001110 0000000011100000 0000000111000000 0000011110000000 0001111100000000 0001111000000000 0111100000000000 0111000000000000 1110000000000000 1100000000000000	Helix-Helix interaction
Four-helical Cytokines	000000010 000001111 000001111 000111100 000110000 001110000 011100000 111000000 110000000	Coil-Sheet interaction

A few significant patterns from the data base *Pattern_DB* are presented here along with the corresponding 3D structural motifs. PyMol [6] software is used to find out the corresponding 3D substructures. Further, we studied the literature to understand the biological significance of the structural pattern and present the same here quoting the relevant references.

Table 5.2: The distribution of proteins among the six folds in All-Alpha class

Fold Name	Fold Index	Training Instances	Testing Instances
Globin-like	1	11	8
Cytochrome -C	3	11	5
DNA-binding 3-helical bundle	4	18	10
4-helical up-and-down bundle	7	8	7
4-helical cytokines	9	12	10
EF-hand	11	8	8
Total		**68**	**48**

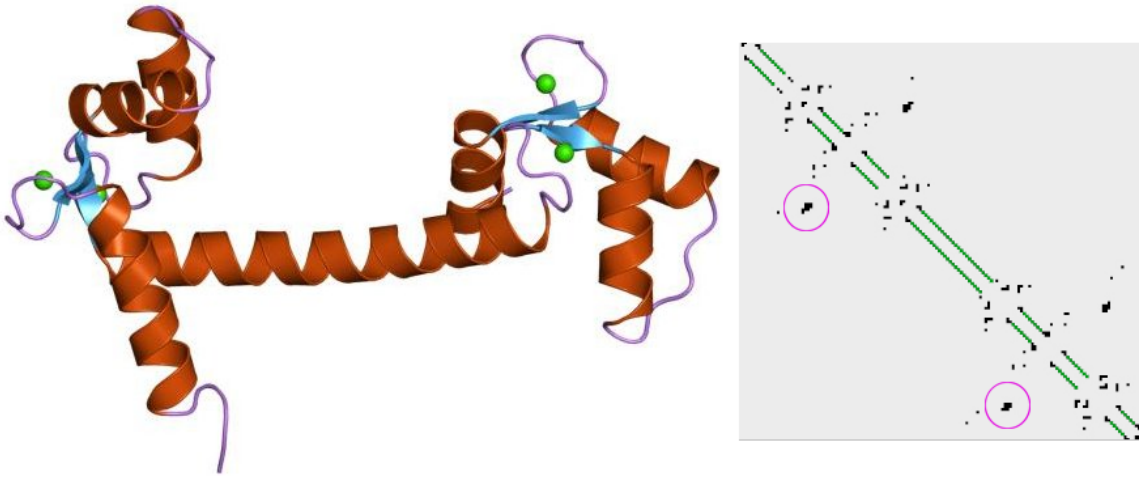


Figure 5.2: The typical orthogonal pair of helices is seen attached by a short parallel beta sheet. The region connecting the helices gives rise to a pattern significant to EF-Hand fold.

5.4.1 EF-Hand fold

The EF-hand motif contains a helix-loop-helix topology, in which the Ca2+ ions are coordinated by ligands within the loop [37]. It consists of two alpha helices positioned roughly perpendicular to one another and linked by a short loop region (usually about 12 amino acids) that usually binds calcium ions. The interactions between the short anti-parallel beta sheets are circled in the Figure 5.2, and the corresponding 2D pattern is shown in the Figure 5.3. We quote [37], "... the EF-hand consists of a nine-residue entering helix, a nine-residue loop and an 11-residue exiting helix. An additional secondary-structural element is observed in the pair as a small β-sheet formed between residues in the latter part of the loops, through which passes the pseudo-2-fold axis of symmetry that relates the EF-hand pair, in most cases the functional unit of Ca2+ binding (PDB code 1EXR)". Clearly it is quite exciting to retrieve a significant pattern through the pattern mining algorithm.

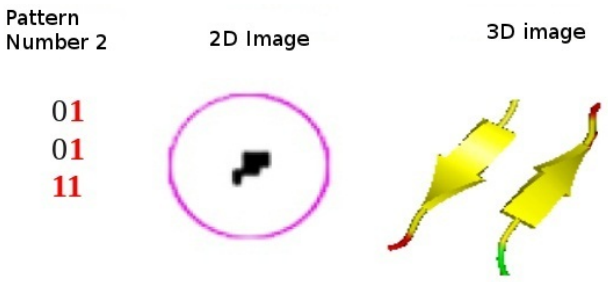


Figure 5.3: 1OSA: EF-Hand-Fold Specific Pattern

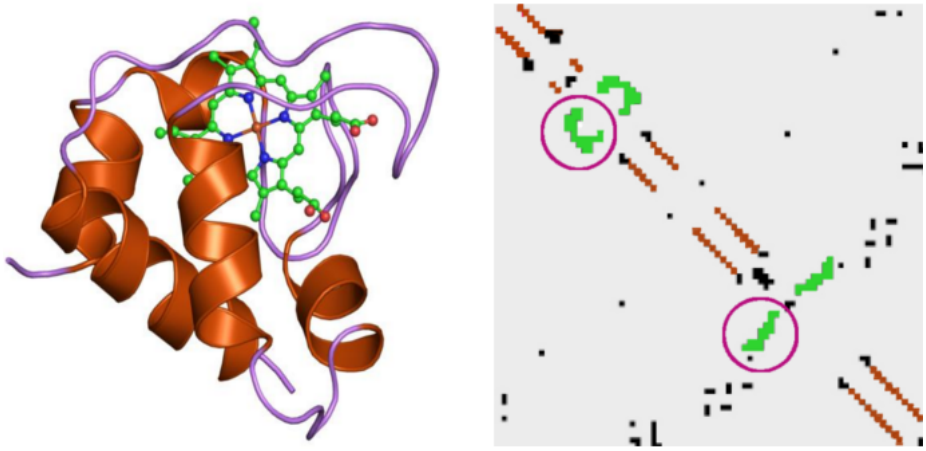


Figure 5.4: 351C: Cytochrome -C Fold 3D structure and 2D Contactmap

5.4.2 Cytochrome-C fold

Two significant patterns of Cytochrome -C fold are shown in the Figures 5.4, 5.5. We find that the 'C'-like pattern in Pattern 27 corresponds to the well-known Cytochrome -C heme group which is an organic molecule, arranged in a circle, with an iron atom in the middle [88, 28, 78, 7]. The iron atom in collaboration with the rest of the heme molecule is said to induce oxygen attracting properties in the entire molecule. Pattern number 6 corresponds to interactions within a long loop region which is again specific to this fold.

5.4.3 Four helical up and down bundle

The four helix bundle is a common structural motif among natural proteins. In this motif four helices pack is arranged roughly lengthwise. Structurally, the four helix bundle

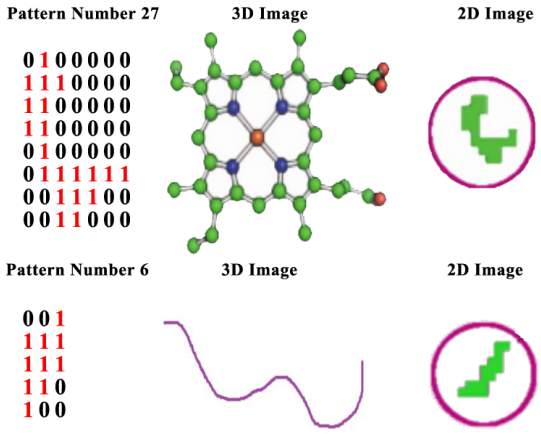


Figure 5.5: 351C: Cytochrome -C Specific Patterns

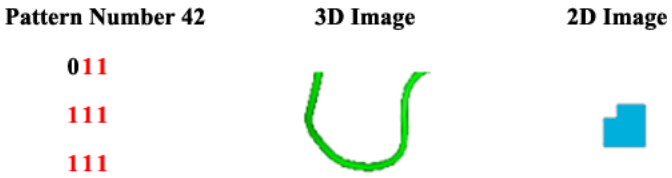


Figure 5.6: Beta hair pin found in 1CGO protein of Four helical up and down bundle

can occur as an isolated fold or as part of a larger protein. Four helix bundles have been observed in numerous medically important proteins such as human growth hormone [79]. The pattern shown in the Figure 5.6 corresponds to the beta hairpin structure in this fold. In Table 5.3, we summarize generic and specific nature of a few of these patterns. For example, pattern 42 is seen only in 4-Helical up and down bundle whereas pattern 4 (P_4) is present in many of the folds in All-Alpha class.

5.4.4 Four helical cytokines

The Four-helix bundle takes an up-up-down-down fold, which forms two-layer packing of anti parallel helix pairs. Figure 5.7 shows the four helical cytokines structure in this fold. The pattern11 is present in the contact maps of four helical cytokines but not in four helical up and down bundle as seen in Table 5.5. This fold consists of four anti parallel alpha helices, termed A, B, C, and D, that are connected by two long crossover links, AB and CD, as well as a short loop, CD, arranged in a left handed twisted helical. Leptin consists of a four-helix bundle similar to the long chain helical cytokines family. Long-chain helical cytokines include granulocyte colony-stimulating factor (G-CSF), leukemia inhibitory factor (LIF), ciliary neurotrophic factor (CNF), and human growth factor (hGF).

In Table 5.3 rows represent frequency of patterns P_k in a fold. Column gives frequency

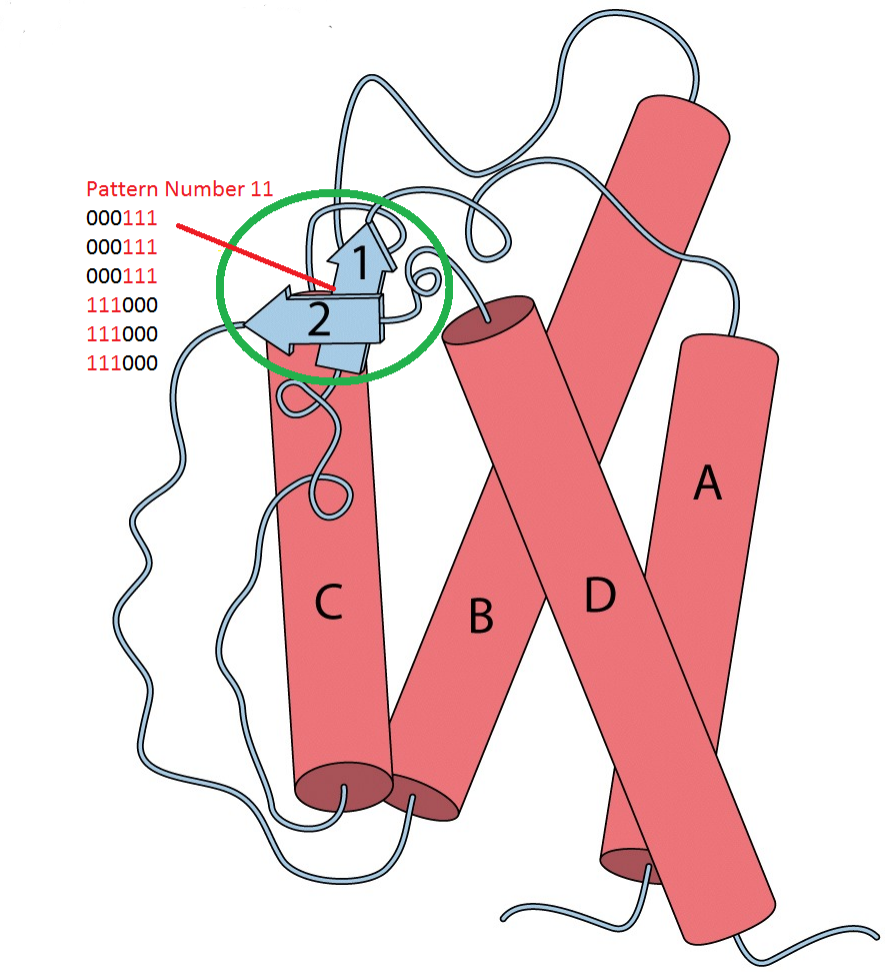


Figure 5.7: Four helical cytokines

Table 5.3: The frequency of occurrence of specific patterns in the proteins of test data set in order to show the specificity the patterns for a fold

Fold Name	P_2	P_3	P_4	P_6	P_8	P_{11}	P_{27}	P_{42}
Globin(8)	-	-	5	-	-	-	-	-
Cytochrome-C(5)	-	3	-	5	-	-	4	-
DNA/RNA binding 3-helical bundle(10)	-	5	-	-	-	-	-	-
Four helical up-and-down bundle(7)	-	-	4	-	-	-	-	4
Four helical cytokines(10)	-	-	3	7	5	3	-	-
EF-hand(8)	8	-	4	-	-	-	-	-

of the particular pattern among different folds. A combination of patterns P_3,P_6 and P_{27} are present in all the proteins of Cytochrome -C fold. It can be seen that pattern P_2 is specific to EF-Hand fold and similarly P_{27} is specific Cytochrome -C fold. Finally the generic patternP_4 is present in globin like, four helical up and down bundle, four helical cytokines and EF-hand folds in All-Alpha class. Clearly, it seems that a few patterns are highly frequent for a particular fold and the approach of frequent-item set mining seems very appropriate.

5.5 Protein structural class and fold recognition

It is clear that these 2D patterns may turn out to be useful features in order to distinguish the different folds of proteins. Protein fold classification is a challenging problem with the best average accuracy reported in the literature being approximately 60.5% [69] on the data set of Ding et al.

5.5.1 Feature vector representation

Every protein contact map is represented as $(n + 6)$ feature vector, where n is the total number of significantly dissimilar patterns obtained from *ExtractPatterns* algorithm and additionally six features that correspond to the secondary structure information lying along the diagonal.

1. **Diagonal features**: Secondary structural features from the diagonal region of the contact map are extracted using the algorithm *Extract_SSP* given in chapter 3: number of helices, minimum and maximum helix length, number of beta strands, minimum and maximum beta length.

2. **Significant patterns**: The off-diagonal interactions which are extracted as 2D patterns using the *Sig-Pattern* algorithm are taken as features. We work with the 126 significantly different 2D-pattern features obtained from the proteins.

3. **Feature set**: We combine diagonal and pattern features to constitute the feature set. A snapshot of the feature vector table is given in Table5.4

Table 5.4: Feature Vector Table

Protein ID	P1	P2	$\cdots$	P_n	No. of helices	min helix length	max helix length	No. of beta strands	min beta length	max beta length
351C	1	0	...	1	10	3	19	5	3	9
1SW8	0	1	...	1	9	3	8	3	3	10
1OSA	0	1	...	0	8	3	10	5	3	12
1BBH	0	0	...	0	5	3	6	2	3	5
1CGO	0	0	...	1	3	3	6	4	3	7
3MDD	1	1	...	0	3	3	10	3	3	7

Since our aim is to obtain 'signature' of each fold, we adopt the framework of class based association rule-mining developed by Liu et al. [54]. CBA-rule generator(CBA-RG) generates frequent rule items. By setting the thresholds for support and confidence, class association rules are generated which are used for fold classification.

5.5.2 Classification algorithm

We use the standard parameters of confidence and support for rule generation. Support of a rule $A \rightarrow B$ is the number of instances for which the rule is satisfied and confidence of the rule $A \rightarrow B = \frac{Support(A \rightarrow B)}{Support(A)}$

We use the well-studied Class Based Association (CBA) Rule Mining algorithm in order to extract rules for classification [8]. Frequent pattern mining is a core procedure within the CBA algorithm.

Step 1: Prepare the pattern vector table for the entire data set.

Step 2: Divide the table into training set and test set in the ratio of 80:20.

Step 3: Apply Apriori algorithm on the training set of feature vectors given in Table 5.4 to generate frequent item sets.

Step 4: Generate the item-set rules which satisfy confidence of 30% and having a minimum support of 2 in the training set.

5.5.2.1 Frequent item-sets

We present here an intermediate result of Apriori algorithm which generates frequent item sets. We list the frequent 3-item-sets that are obtained by the algorithm in Table 5.5

Table 5.5: Frequent 3-item-sets obtained in All-Alpha class

All-Alpha Class	3-Item-sets
Globin	Pno71=1,Pno3=0,Pno4=1 Pno8=0,Pno5=0,Pno1=1
Cytochrome C	Pno6=1,Pno40=0,Pno1=1 **Pno27=1,Pno6=1,Pno4=0** Pno42=0,Pno3=1,Pno2=0
DNA/RNA binding 3-helical bundle	Pno32=0,Pno14=1,Pno7=0 Pno74=1,Pno40=0,Pno3=0 Pno46=1,Pno27=0,Pno4=0
4-helical up-and-down-bundle	Pno30=1,Pno8=0,Pno5=0 Pno6=0,Pno4=1,Pno1=0 **Pno42=1,Pno7=0,Pno5=0**
4-helical cytokines	Pno40=1,Pno30=0,Pno2=0 **Pno6=1,Pno4=1,Pno3=0** Pno11=1,Pno8=1,Pno4=1
EF-Hand	**Pno74=0,Pno40=0,Pno2=1**
Generic	**Pno6=1,Pno4=1,Pno3=0**

A few of the 3-item-sets that correspond to the patterns discussed in the earlier section are highlighted in Table 5.5. Cytochrome-C fold contains patterns Pno27 and Pno6 as shown in 5.5 that got extracted as a frequent item set for this fold. Beta hair pin represented as Pno42 in Figure 5.6 is found to be a conserved substructure in Four helical up and down bundle. A pattern that is conserved as a whole among the proteins correspond to generic patterns like the loop structure as given in 5.5 represented as Pno6.

5.5.2.2 Rule mining

The association rule mining algorithm CBA generates rules with predicate of features on the left hand side and class label on the right hand side Minimum helix (beta) length and maximum helix (beta) length are denoted in the rules as min_helix (beta) and max_helix (beta) respectively. Number of helices and beta strands are denoted as no_of_helices (betas) respectively. For example, a rule is of the form

min_beta=3, no_of_helices =9 → class= 1.

The globin fold is found in its namesake proteins hemoglobin and myoglobin as well as in phycocyanin proteins. As per Turcotte et al.[80] the maximum number of helices that a globin fold has 8. In our classification system we obtain a similar rule as minimum beta as 3 and number of helices is 9. Globin fold is a bundle of 8-helices linked by relatively short connecting loops. As per literature [15] this fold typically consists of eight alpha helices but, some proteins may also have additional helix extensions at their termini.

Turcotte et al. derive the number of helices in ILP rule as $(3 =< A =< 3)$ for DNA/RNA binding 3-helical bundle our classification rule predicts the number of helices as (number of helices = 3 || number of helices =4). The four helices may be arranged in a simple up-and-down topology. Additionally the number of betas is predicted to be with 3 the maximum beta length as 5.

Now the model of rule-miner is trained and the rules are applied to the test set of Ding et al. [29]. The rules are evaluated for their classification performance on all the 27 folds and for each of the four structural classes.

5.6 Results of fold classification

The top association mining rules satisfying the confidence and support thresholds are applied for classification and the performance is evaluated with the standard measures of Precision, Recall and F-Measure whose definitions are given in Section 4.4.1. We compare the performance of this classifier with those of Shamim et al. [69] and Ding et al. [29] who report classification on the data set of Ding et al. During comparison, we give only results for Recall as only these are available for the classification results given by Ding et al.

The classification results obtained are tabulated in Tables 5.7, 5.8 and 5.9 for structural class and fold prediction. Clearly, better classification accuracy is achieved by using the feature set which uses both the diagonal SSE's and the off-diagonal pattern features as shown in Table 5.9. Also, note that the results for Recall obtained are better in most cases when compared to the best in the literature, those of Ding et al. [29] and Shamim et al. [69] who use high dimensional feature vectors composed of both structural features and the sequence amino acid composition features.

5.6.1 Fold 'signatures': A sample

- Rule1: no_of_betas = 1,Pno2=1 $\rightarrow$ class = 11
- Rule2: max_beta_length = 7, no_of_helices = 10 $\rightarrow$ class = 11

Table 5.6: Some significant rules in All-Alpha class

All-Alpha Class	Significant Rule
Globin	Rule1: min_beta=3, no_of_helices=9 → class=1 Rule2: min_helix = 5 → class = 1 Rule3: Pno3 = 0, max_helix = 19 → class = 1 Rule4: max_beta = 4, no_of_betas = 3 → class = 1 Rule5: Pno11 = 0, no_of_helices = 8 → class = 1
Cytochrome -C	**Rule1: Pno6 = 1, Pno27 = 1 → class = 3** Rule2: Pno1 = 0, no_of_betas = 13 → class = 3 Rule3: Pno2 = 0, Pno94 = 1 → class = 3 Rule4: Pno9 = 1, Pno30 = 1 → class = 3 Rule5: Pno1 = 1, Pno2 = 0 → class=3 Rule6: no_of_helices = 5, max_helix = 14 → class = 3 Rule7: Pno47 = 0, Pno10 = 1 → class = 3 Rule8: min_helix = 3, no_of_helices = 5 → class = 3
DNA/RNA binding 3-helical bundle	Rule1:Pno27 = 0, Pno2 = 0, no_of_helices = 3→class = 4 Rule2:Pno8 = 1, max_beta = 10 →class = 4 Rule3:Pno17 = 1, Pno37 = 1→ class = 4 Rule4:Pno7 = 0, Pno87 = 1 → class = 4 Rule5:Pno71 = 0, Pno11 = 0, no_of_helices = 4 → class = 4 Rule6:max_helix = 16 →class = 4 Rule7:Pno8 = 0, Pno17 = 1 → class = 4
4-helical up&down-bundle	**Rule1:Pno42 = 1, no_of_helices = 6 → class = 7** Rule2:no_of_helices = 5, no_of_betas = 3 →class = 7 Rule3:Pno11 = 0, max_beta = 5 → class = 7
4-helical cytokines	Rule1:Pno43 = 0, Pno97 = 1→class=9 Rule2:no_of_betas=7, no_of_helices=7→class = 9 Rule3:Pno11 = 1, no_of_helices = 6→class = 9 Rule4:no_of_helices = 5, max_helix = 24→class = 9 Rule5:Pno11 = 1, max_helix = 24→class = 9
EF-Hand	**Rule1:no_of_betas = 1, Pno2=1→class = 11** Rule2:max_helix = 11, no_of_helices = 8→class = 11 Rule3:max_beta = 7, no_of_helices = 10→class = 11 Rule4:min_helix = 3, max_beta = 7→class = 11

- Rule3:Pno42 = 1,no_of_helices = 6 → class = 7
- Rule4: Pno6 = 1,Pno27 = 1 → class = 3
- Rule5: min_beta=3, no_of_helices=9 → class= 1

In Figure 5.8, we show the classification results obtained within the All-Alpha class.

Table 5.7: Structural classification of a protein into the four structural classes (Diagonal features)

Class	Precision %	Recall %	F-Measure
All-Alpha	83	70	75.9
All-Beta	80	80	80
Alpha/Beta	75	90	81.8
Alpha+Beta	81	43	56.1
Average	**79.75**	**70.75**	**74.45**

Table 5.8: Structural classification of a protein into the four structural classes (Pattern features)

Class	Precision %	Recall %	F-Measure
All-Alpha	55	83	66
All-Beta	76	83	79
Alpha/Beta	86	77	81
Alpha+Beta	100	26	41
Average	**79.25**	**67.25**	**66.75**

Table 5.9: Comparison of accuracy for structural classification with other methods using (**Diagonal+Pattern**) features

Class	Precision %	Recall % (Accuracy)	F-Measure	Shamim% [69] (Accuracy)	Ding[29]% Accuracy
All Alpha	78	86.5	82.03	-	-
All Beta	90	71.4	79.62	-	-
Alpha/Beta	76	78.26	76.12	-	-
Alpha+Beta	80	49.9	61.46	-	-
Average	**81**	**71.55**	**74.8**	**60.5**	**51.1**

It is clear that our algorithm performs quite well on all the folds except on Globin fold. Clearly the pattern features of Cytochrome -C and EF-Hand as discussed in Section 5.4 seem to be useful in distinguishing the specific fold as well as to differentiate among the many folds within All-Alpha class.

The total data set of Ding et al. is given in Table A.1 in the Appendix and the 27-way Fold classification results using feature set which uses both the diagonal SSE's and the off-diagonal pattern features are shown in Table A.2. The specific fold of globin like fold of All-Alpha class is not classified well when compared to the existing literature. In globin like fold, according to the literature, majority of helices are present and hence we have only diagonal patterns and no off- diagonal patterns which may be reducing the accuracy. Our

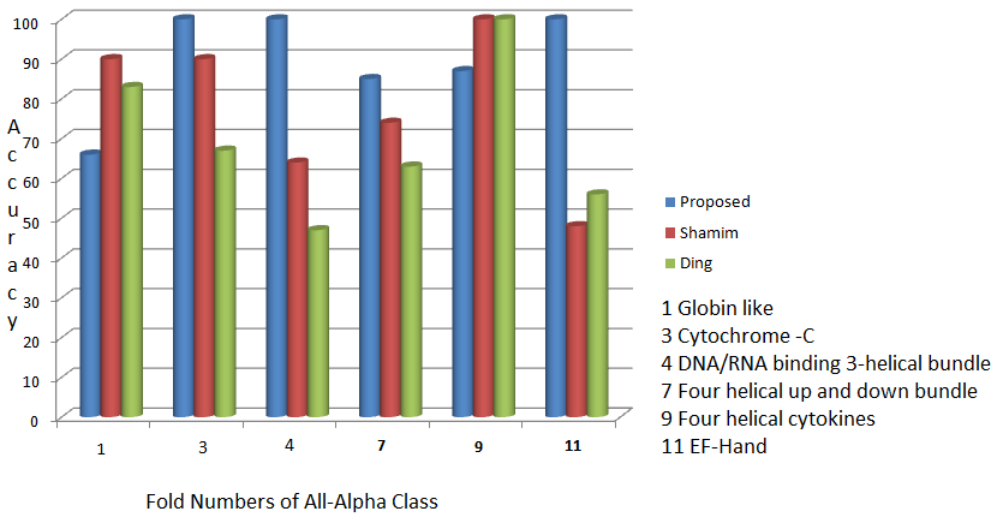


Figure 5.8: Classification within All-Alpha class using diagonal+pattern features and comparing accuracy with existing methods

study shows that some folds are consistently recognized well with high prediction accuracy in All-Alpha class: folds of Cytochrome -C, Four helical up and down bundle and EF-hand as depicted in Figure 5.8. Further in our experimentation we found that in the All-Beta class, both the folds of Viral coat and OB fold are recognized well and in Alpha/Beta class except one fold, all the remaining folds are recognized with high accuracy. Alpha+Beta class has only three folds and among these, the fold of small inhibitors is classified with high accuracy, but performed poorly on the rest making the accuracy a very low value of 43%.

Further, we find that three folds, namely, cuperedoxins fold, ribonuclear H-like motif and beta-grasp fold belonging to All beta, Alpha+Beta and Alpha/Beta classes respectively are not at all recognized. These folds are known to contain very small beta sheets and the diagonal features may be mis-representing these sheets as turns which in turn is affecting the accuracy. We need to conduct a deeper study regarding the pattern features within these folds and analyze the poor performance.

Now we test the classifier on the predicted contact maps of the test set of All-Alpha class in order to validate our whole approach as an ab-initio method. We present the results in Table 5.10.

Table 5.10: Accuracy results on predicted contact maps obtained for All-Alpha fold classification

Fold Name	Shamim[69] %	Ding[29]%	Proposed method
Globin-like	90	83	75
Cytochrome -C	90	67	100
DNA/RNA-binding 3-helical bundle	64	47	80
4-helical up and down bundle	74	63	86
4-helical cytokines	100	100	80
EF-Hand	48	56	88
Average	**77.66**	**69.33**	**84.8**

5.7 Results on predicted contact maps

The CBA rule classifier has been trained using contact maps built using 3D structural information of the proteins. This classifier is supplied with the test set of predicted contact maps. The test data set of All-Alpha class containing 48 sequences is submitted to *Distill* software to obtain the contact maps. We recall that these contact maps have been predicted from the amino acid sequence and do not use 3D structural information. In order to validate our approach as as ab-initio method, we test the classifier on the set of predicted contact maps. Firstly, note that the accuracy on All-Alpha class with the original contact maps is 86.5%(see Table 5.9) which is very close to the result of 84.8% on predicted contact maps. The results in Table 5.10 clearly show that the rule based classifier is performing in most cases superior to the existing classifiers.

5.8 Conclusion

Novel features have been obtained by mining protein contact maps using a simple and effective pattern extraction algorithm. We extend the *ExtractPatterns* algorithm by ensuring retrieval of patterns with minimum dissimilarilty. *Sig_Pattern* algorithm is shown to be quite effective in extracting patterns that are significant for a fold. Further, we demonstrate correspondence of these patterns to specific locations in the tertiary structure whose importance is then collated with literature related to protein structure and presented here. The effectiveness of these features is tested by using these to carry out fold recognition which is a challenging multiclass classification problem. We show that using class based association rule mining algorithm, a 'signature' can be composed for each class. The details regarding the accuracy of signatures for folds in the All-Alpha structural class are given and shown to predict with an accuracy of nearly 90%. The same process is carried out for all the four

structural classes and the results of performance are given for the 27-way fold classification. Further the classifier is tested on the set of predicted contact maps and is shown to perform with higher accuracy than the existing methods. Overall accuracy for the 27-way fold classification with this approach came out to be 71.55% much better than even that obtained by boosting technique in the previous chapter. Alpha+Beta structural class poses challenge which needs to be investigated further.

Chapter 6

Application: Contact Map Overlap Problem

6.1 Introduction

Protein structure comparison is one of the most challenging problems in bioinformatics. This problem is modeled as a contact map overlap problem in which the similarity of the two proteins being compared is measured by the amount of 'overlap' between their corresponding protein contact maps. To find a maximum overlap is proved to be an NP-Hard problem [60]. There are several solutions proposed including integer programming techniques [20], approximate [39, 11] and heuristic algorithms [20, 60, 19].

6.2 Literature

The first exact algorithm for the maximum contact map overlap(Max-CMO) problem, based on integer programming(IP), was developed Lancia [52] et al. and improved by Caprara in [20] et al. Later, several other methods based on the same approach were proposed [12, 75, 93]. Agarwal et al. [11] improved the approximation ratio of Goldman et al. [39] modeling similarity of two polygon chains as a graph-theoretic problem. Recently a polynomial approximation scheme for the contact map alignment problem has been developed [94]. Several of the recent approaches adopt alignment of eigen vectors to solve the contact map overlap problem [61, 72, 73].

Many heuristics have been proposed for contact map overlap(CMO) problem which achieve excellent performance on bench-mark data sets. Pelta et al. [60] propose a variable

neighborhood search meta heuristics algorithm for solving Max-CMO. An algorithm called Bimal [19] has been proposed which uses local weight functions as heuristics to construct a weighted bipartite graph. They use a procedure called Maximum Non Crossing Matching (MNCM) of Malucelli et al. [55] to compute the alignment which gives the overlap.

We propose to build a divide and conquer approach to addresss contact map overlap problem. We extract the region of contacts, which may themselves be treated as small contact maps. Using *Approximate 2D-Pattern Matching* algorithm we find a matching region in the other contact map, then use one of the heuristics internally between each pair of matched regions to construct an alignment. Finally use MNCM to merge the region-wise alignments. An obvious advantage of this algorithm is that it facilitates parallelization. To the best of our knowledge, this kind of approach to CMOP has not been proposed in the literature.

6.3 Motivation

The last few chapters have shown that we can extract 2D-patterns from the off-diagonal region of the contact map. As depicted in Figure 6.1, if two proteins are similar, can we align the corresponding 2D-clusters such that the contact map overlap problem be addressed on the basis of these 'matching' clusters?

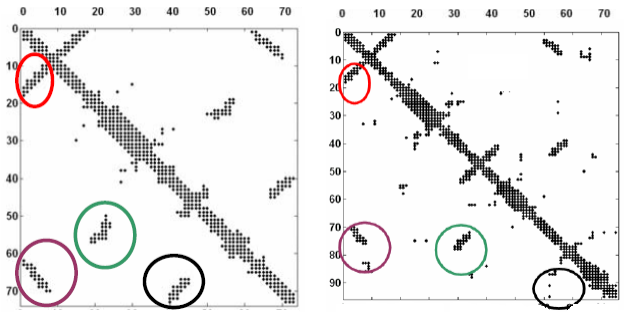


Figure 6.1: Can we find an alignment that maximizes the matching of the circled patterns of same colour in the two proteins

6.4 Problem definition

A, B be two proteins whose contact graphs are $A = (V_1, E_1)$ and $B = (V_2, E_2)$ where $V_1 = \{i_1 < i_2 < i_3 < \cdots < i_n\}$ is an ordered set of vertices and

$E_1 \subseteq \{(i_k, i_l), 1 \leq k, l \leq n\}$ of edges and
$V_2 = \{j_1 < j_2 < j_3 < \cdots < j_m\}$ and $E_2 \subseteq \{(j_k, j_l), 1 \leq k, l \leq m\}$.

- To find alignment mapping function $f : V_1 \rightarrow V_2, \ni f$ is $1 \rightarrow 1$ and having non crossing matching property. $i_k < i_l \Rightarrow f(i_k) < f(i_l)$.
- Overlap of f consists of common edges $(i_p, i_q) \in E_1, (j_r, j_s) \in E_2 \; where$ $f(i_p) = j_r, f(i_q) = j_s$.

The problem is to find an alignment mapping f that maximizes the overlap between the two proteins.

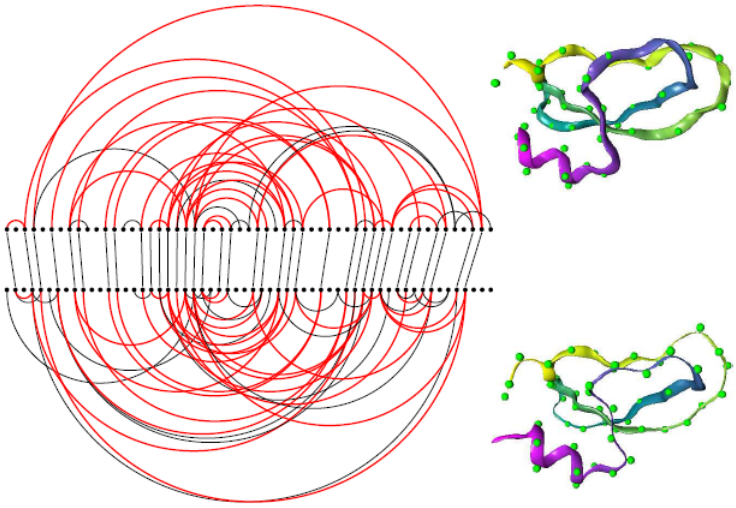


Figure 6.2: Representation of two protein graphs depicting CMO [52]

In Figure 6.2 on the left, two proteins A and B are depicted as graphs and their corresponding 3D structures are given on the right. A consists of 55 residues with 43 contacts(edges) and B consists of 58 residues with 53 contacts. The Figure 6.2 shows black color arcs representing an overlap of 31 overlap edges.

6.5 Methodology

We extract patterns using *ExtractPatterns* algorithm from the contact maps of two proteins A and B. A matching score is computed between all pairs of patterns of the two contact maps using an *Approximate 2D-Pattern Matching* algorithm. Pairs of matched patterns

with optimal score are obtained by using dynamic programming approach. These are themselves pairs of small contact maps. Thus we have evolved a divide and conquer strategy in which a pair of large contact maps have been divided into pairs of small contact maps. We apply MSVNS heuristic algorithm to the small contact maps and obtain alignments. Finally the results so obtained are 'merged' using maximum non-crossing matching technique to compute the overall alignment. This alignment is used to obtain the final overlap.

6.5.1 Divide and conquer procedure

Step 1 ExtractPatterns(A)
/* Returns patterns PA_i of A, $0 \leq i \leq np(A)$ */
ExtractPatterns(B)
/* Returns patterns PB_j of B, $0 \leq j \leq np(B)$ */

Step 2 For each PA_i in A, $\quad i = 1 \cdots np(A)$
For each PB_j in B, $\quad j = 1 \cdots np(B)$
Approximate 2D-Pattern Matching(PA_i, PB_j)
Return normalized match score (PA_i,PB_j)

Step 3 Apply dynamic programming alignment algorithm between
(PA_i, PB_j), $1 \leq i \leq np(A)$, $1 \leq j \leq np(B)$
Return maximally matched pairs (PA_k, PB_l) $1 \leq k \leq np(A)$ and $1 \leq l \leq np(B)$

Step 4 MSVNS (PA_k, PB_l)
/* It returns alignment between PA_k and PB_l along with overlap */

Step 5 Merge the local alignments to get a global alignment between A and B.

The flow of the algorithm is given in Figure 6.3. To understand the algorithm consider the Figure 6.4 that depicts contact maps of the proteins 1B00(122) and 4TMYA(118) both of which belong to Flavodoxin-like fold according to SCOP classification. In Figure 6.4 regions circled represent the patterns. The contact map on the left has seven regions and marked with different colored circles and the right contact map also has the same number of colored circles. Instead of aligning the two contact maps globally, we plan to align

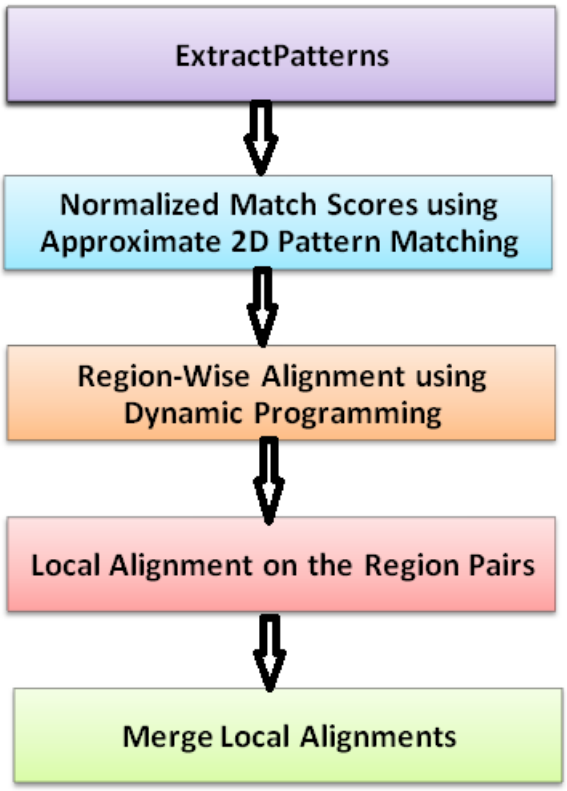


Figure 6.3: Representation of flow

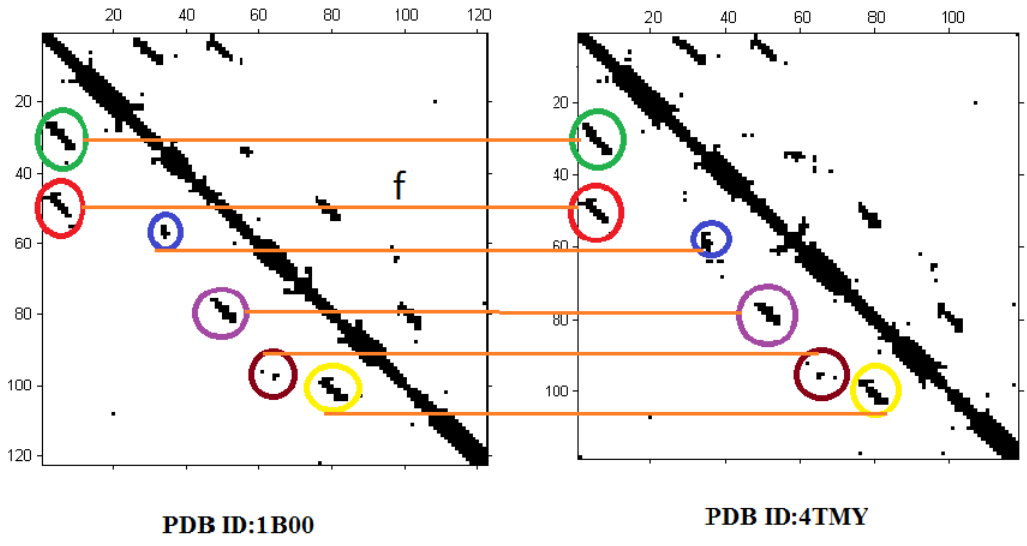


Figure 6.4: Two proteins of Flavodoxin-like fold with similar pattern configurations seen in their corresponding contact maps

these regions using *Approximate 2D-Pattern Matching* algorithm and dynamic programming method. Since protein structure alignment should correspond to common/similar protein substructures being aligned, we would like to utilize the alignment between clusters thus obtained for the overall contact map alignment problem.

6.6 Algorithm and implementation details

Implementation of the procedure is carried out on the bench mark data set due to Skolnick et al. [9]. Many authors of CMO evaluate the performance of their algorithms on this data set [60, 19, 72].

6.6.1 Data set

The bench-mark data set of Skolnick et al. is given in Table 6.1. It contains 40 small and medium size proteins having number of residues between 97 and 255 distributed among five SCOP families.

Table 6.1: Skolnick data set

Name of Fold	Number of proteins	Name of the proteins
Flavodoxin-like	11	1b00, 1dbw, 1nat, 1ntr, 3chy, 1qmp(A,B,C,D),4tmy(A,B)
Cupredoxin-like	9	1baw,1byo(A,B),1kdi, 1nin, 1pla, 2b3i, 2pcy, 1plt
Tim beta/alpha-barrel	11	1amk, 1aw2, 1b9b, 1btm, 1hti, 1thm, 1tre, 1tri, 1ydv, 3ypi, 8tim
Ferritin-like	6	1b71, 1bcf, 1dps, 1fha, 1ier, 1rcd
Microbial ribonuclease	3	1rn1(A,B,C)

6.6.2 Extraction of patterns

ExtractPatterns algorithm is run by setting the parameter of density $d = 4$ on Skolnick data set. The number of contacts in a pattern denotes the density of the pattern. Table 6.2 gives the details with a total of 517 patterns being obtained with a minimum density of 4 and maximum density being 414.

6.6.3 Approximate 2D-pattern matching

Consider two protein contact maps A and B. Let the set of patterns obtained from first protein contact map A be $A = \{PA_1, PA_2, PA_3, \cdots PA_m\}$ and those from B be $B =$

Algorithm 5 Approximate 2D-Pattern Matching

Input: Text[M][N], Pattern[m][n]
Output: Return maximum pattern count value
Input parameters: density, match, threshold

```
density← 0
for i ← 0 to M-1 do
    for j ← 0 to N-1 do
        if Text(i,j) == 1 then
            density = density +1
        end if
    end for
end for
threshold ← density * 0.66
𝕄ℂ ← ∅
for i ← 0 to M-1 do
    for j ← 0 to N-1 do
        match ← 0
        for r ← 0 to m-1 do
            for s ← 0 to n-1 do
                if Text(i + r, j + s) == 1 then
                    if Pattern(r, s) == 1) then
                        match = match+1
                    end if
                end if
            end for
        end for
        if match >= threshold then
            x ← i + r -1
            y ← j + s -1
            𝕄ℂ ← {𝕄ℂ}∪{(x,y,match)}
        end if
    end for
end for
Return maximum match value from 𝕄ℂ
```

Table 6.2: Extraction of patterns

Fold Name	No.Patterns	Min density	Max density
Flavodoxin-like	116	5	108
Cupredoxin-like	132	5	59
Tim beta/alpha-barrel	216	5	225
Ferritin-like	26	4	414
Microbial ribonuclease	27	5	81

$\{PB_1, PB_2, PB_3, \cdots PB_n\}$.

Approximate 2D-Pattern Matching algorithm is applied to all pairs of patterns (PA_i, PB_j), which gives the scoring matrix $(sv_{i,j}) : 1 \leq i \leq m, 1 \leq j \leq n$

6.6.4 Construction of normalized scoring matrix

It is important to normalize the scoring values since raw scores may be high for larger matrices and does not truly reflect the amount of matching. The normalization of the score is done using the window size of the smaller pattern. If score value between two patterns obtained is sv and smaller of the patterns is of dimension $m \times n, k = min(m, n)$ then, normalized score is calculated as sv/k.

6.6.5 Dynamic programming

Dynamic programming technique is a standard approach to align pairs of patterns given a scoring matrix. In general, there are many possible alignments between any two patterns that it would be terribly inefficient to search through all of them for the best one. We carry out the traditional alignment procedure using dynamic programming approach which simply computes the current maximum by making the decision between considering the new entry or take the previous score which would introduce a gap in the alignment. This procedure fills in the table row by row, and left to right within each row using the recurrence relation given here. Each entry takes constant time to fill in, so the overall running time is just the size of the table, $O(mn)$.

- $S(i, 0) = 0, i \geq 0$
- $S(0, j) = 0, j \geq 0$

$$S(i,j) = Max \begin{cases} S(i-1,j) \\ S(i-1,j-1) + S(i,j) \\ S(i,j-1). \end{cases}$$

6.6.6 Tracing of the approach on an example

A few patterns that have been extracted when *ExtractPatterns* algorithm is run on two proteins 1B00 and 4TMYA are shown below.

$$PA_2 = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \end{pmatrix} PB_2 = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \end{pmatrix}$$

$$PA_5 = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix} PB_4 = \begin{pmatrix} 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$

$$PA_7 = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{pmatrix} PB_5 = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 0 \end{pmatrix}$$

$$PA_{10} = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix} \quad PB_7 = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$

The matching is computed using the *Approximate 2D-Pattern Matching* algorithm between (PA_i, PB_j) and the matching score is calculated by computing the Hamming distance between the matched patterns $S(PA_i, PB_j)$ and is denoted as $sv(i, j)$. That is, this score gives the maximum number of common contacts between the patterns. Implementation of computation of the scoring matrix between all pairs of patterns obtained from two proteins 1B00 and 4TMYA is given as an example here in Table 6.5.

Table 6.3: Scoring matrix of two proteins

Protein Pair 1BOO_4TMYA	PB1	PB2	PB3	PB4	PB5	PB6	PB7	PB8	PB9
PA1	56	0	0	0	0	0	0	0	0
PA2	0	23	0	0	0	0	0	0	0
PA3	0	0	16	0	0	0	0	0	0
PA4	0	0	7	0	0	0	0	0	0
PA5	0	0	0	19	0	0	0	0	0
PA6	0	0	7	0	0	1	0	0	0
PA7	0	0	0	0	7	0	0	0	0
PA8	0	0	0	0	0	24	0	0	0
PA9	0	0	0	0	0	12	0	0	0
PA10	0	0	0	0	0	0	19	0	0
PA11	0	0	0	0	0	29	0	0	0
PA12	0	0	0	0	0	0	0	15	0
PA13	0	0	0	0	0	0	0	0	33

6.6.7 Region-wise alignment using DP method

The numbers colored blue in Table 6.5 show the mapping path using backtracking technique. Here the path calculation starts from the bottom right to the top left corner, when reading out the scores from scoring matrix. If the scores in PA_i and PB_j are equal, they are part of the scores, and we go both up and left. If not, we go up or left, depending on which cell has a highest score. This corresponds to either taking the highest scores between $S(i, j-1)$ and $S(i-1, j)$. The Table 6.5 shows the backtracking method. Based on these scores we identify the pairwise alignment of patterns from the proteins.

Table 6.4: Normalized scores of proteins given in Table6.3

Protein Pair 1BOO_4TMYA	PB1	PB2	PB3	PB4	PB5	PB6	PB7	PB8	PB9
PA1	15.954	0	0	0	0	0	0	0	0
PA2	0	82.14	0	0	0	0	0	0	0
PA3	0	0	57.14	0	0	0	0	0	0
PA4	0	0	46.66	0	0	0	0	0	0
PA5	0	0	0	90.47	0	0	0	0	0
PA6	0	0	19.4	0	0	2.77	0	0	0
PA7	0	0	0	0	100	0	0	0	0
PA8	0	0	0	0	0	43.63	0	0	0
PA9	0	0	0	0	0	21.81	0	0	0
PA10	0	0	0	0	0	0	90.47	0	0
PA11	0	0	0	0	0	21.323	0	0	0
PA12	0	0	0	0	0	0	0	71.42	0
PA13	0	0	0	0	0	0	0	0	91.66

Table 6.5: $S(i, j)$ computed between P1 and P2 using dynamic programming method

Protein Pair 1BOO_4TMYA	Initial_Value	PB1	PB2	PB3	PB4	PB5	PB6	PB7	PB8	PB9
Initial_Value	0	0	0	0	0	0	0	0	0	0
PA1	0	16	16	16	16	16	16	16	16	16
PA2	0	16	98	98	98	98	98	98	98	98
PA3	0	16	98	155	155	155	155	155	155	155
PA4	0	16	98	155	155	155	155	155	155	155
PA5	0	16	98	155	245	245	245	245	245	245
PA6	0	16	98	155	245	245	248	248	248	248
PA7	0	16	98	155	245	345	345	345	345	345
PA8	0	16	98	155	245	345	389	389	389	389
PA9	0	16	98	155	245	345	389	389	389	389
PA10	0	16	98	155	245	345	389	479	479	479
PA11	0	16	98	155	245	345	389	479	479	479
PA12	0	16	98	155	245	345	389	479	550	550
PA13	0	16	98	155	245	345	389	479	550	642

We show an example of the optimal alignment in Table 6.6 obtained using Dynamic programming method on patterns extracted from the two proteins 1BOO and 4TMYA. Table 6.6 shows the pairs of regions from protein A and protein B.

When this algorithm is run on each fold of Skolnick data set, the total number of pairs of patterns that have been obtained is summarized in Table 6.7.

Table 6.6: Alignment between regions of protein contact maps A and B

PA_1	PB_1
PA_2	PB_2
PA_3	PB_3
PA_5	PB_4
PA_7	PB_5
PA_8	PB_6
PA_{10}	PB_7
PA_{12}	PB_8
PA_{13}	PB_9

Table 6.7: Pattern pairs from Skolnick dataset

Fold name	Number of pairs of aligned patterns
Flavodoxin-like	503
Cupredoxin-like	359
Tim beta/alpha-barrel	754
Ferritin-like	29
Microbial and ribonuclease	27

6.6.8 Alignment of smaller contact maps residues using MSVNS algorithm

We choose the heuristic algorithm of Pelta et al. [60] for carrying out the local alignment between each pair of smaller contact maps in order to obtain alignment at the residue level. Pelta et al. proposed three versions of Multistart VNS(MSVNS), corresponding to different parameter settings on the neighborhood structures. Among these three versions, MSVNS3 proves to be the best choice for our experimentation and we choose MSVNS3 with setting the parameter of window size to 10%.

Each pair of regions (patterns) obtained from the dynamic programming algorithm actually corresponds to two small contact maps from within the original contact maps. These pairs of smaller contact maps are submitted to Multistart Variable Neighborhood Search(MSVNS) algorithm which finds out the maximum overlapping at the residue level. An example is shown in Table 6.8

6.6.9 Merge Algorithm

Clearly the different alignments given by MSVNS on all the pairs of regions lead to conflicts. We simply take the union of all these local alignments to obtain a bipartite graph between residues of proteins A and B. We apply the well-known Maximum Non-Crossing

Table 6.8: MSVNS aligned residues

PA_1	PB_1
1	2’
2	4’
3	5’
4	6’
5	7’
6	8’
7	9’

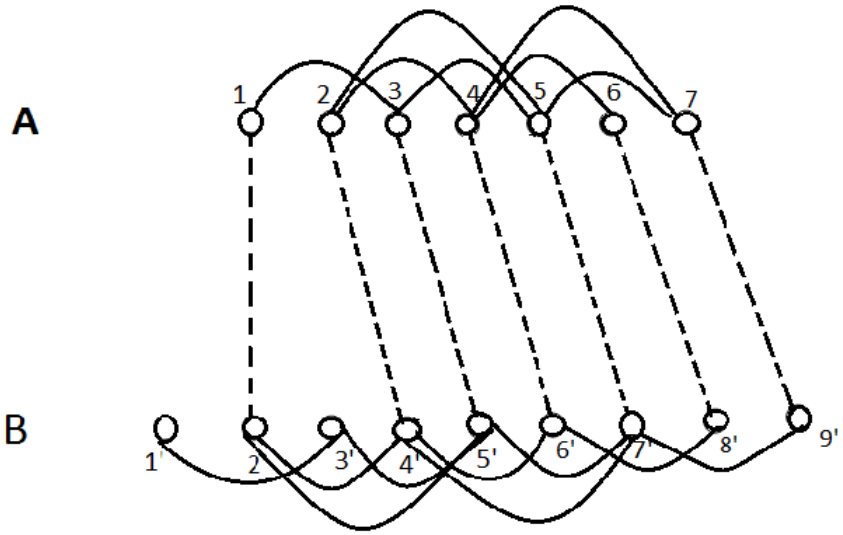


Figure 6.5: MSVNS gives a residue level alignment between a pair of patterns which is depicted here. The amount of common overlap for this alignment is 6.

Matching (MNCM) algorithm of Malucelli et al. [55] on the bipartite graph in order to remove ‘crossing’ edges to get a one-one mapping that gives the maximum matching possible. For completion sake, the algorithm of MNCM is given in Algorithm 6.

6.6.10 Computing the overlap

Overlap of f consists of common edges $(i_p, i_q) \in E_1, (j_r, j_s) \in E_2 \; where f(i_p) = j_r, f(i_q) = j_s$.

For a given alignment mapping $f : V_1 \rightarrow V_2$, of the residues of proteins A and B respectively, the overlap is calculated by simply counting the number of cycles of length four. That is, suppose $f(a_1) = b_1$ and $f(a_2) = b_2$ with both edges $(a_1, a_2) \in E_1$ and

Algorithm 6 NonCrossingMatching(NCM)

Input: Bipartite Graph
Output: Non crossing matching edges
Input parameters: Vertices V_1, Vertices V_2, Edges E

for each node j $\in V_2$ **do**
  lable(j) $\leftarrow$ 0
end for
for i $\leftarrow$ 1 to $\mid V_1 \mid$ **do**
  $\mathbb{N} \leftarrow$ neighbors of i in ascending order
  for each node j $\in \mathbb{N}$ **do**
    e $\leftarrow$ (i, j)
    edgelabel(e) $\leftarrow$ 1 +max{ label(k), $k < j$}
  end for
  for each node j $\in \mathbb{N}$ **do**
    e $\leftarrow$ (i, j)
    label(j) $\leftarrow$ max{ edgelabel(e), label(j) }
  end for
end for
$\mathbb{NCE} \leftarrow \emptyset$
k $\leftarrow$ max{ edgelabel(e), e$\in$E }
$e_k \leftarrow$ argmax{ edgelabel(e), e$\in$E }
$\mathbb{NCE} \leftarrow \mathbb{NCE} \cup \{e_k\}$
k $\leftarrow$ k - 1
while $k > 0$ **do**
  $\mathbb{N}_k \leftarrow$ edges with label k
  for each edge e $\in \mathbb{N}_k$ **do**
    if Crossededges (e, e_k) == FALSE **then**
      $e_k \leftarrow$ e
      $\mathbb{NCE} \leftarrow \mathbb{NCE} \cup \{e_k\}$
      break
    end if
  end for
  k $\leftarrow$ k - 1
end while
Return $\mathbb{NCE}$

$(b_1, b_2) \in E_2$, then the cycle is composed of $(a_1, b_1), (b_1, b_2), (b_2, a_2), (a_2, a_1)$. Note that the edges are undirected and hence $(a_2, a_1) = (a_1, a_2)$ etc.

Algorithm 7 Procedure for overlap

Input: Contact maps of $A = (V_1, E_1)$ and $B = (V_2, E_2)$; Alignment $f : V_1 \rightarrow V_2$
Output: overlap(A, B)

```
overlap = 0
for each edge (a_i, a_j) ∈ E_1 do
  if f(a_i) = b_k and f(a_j) = b_l then
    if (b_k, b_l) ∈ E_2 then
      overlap = overlap + 1
    end if
  end if
end for
```

MSVNS algorithm constructs a correspondence between residues of pattern PA_1 of the contact map of protein A to residues of pattern PB_1 of contact map of B, which is given in Figure 6.5. Residues 1, 2, 3, 4, 5, 6 and 7 in the upper protein (A) are paired with residues 2', 4', 5', 6', 7' ,8' and 9' in the lower one (B). The alignment is represented by dotted lines while the protein contacts are shown with solid ones. In the example we can find 6 cycles. Some of the cycles are one composed by the arcs $(1, 3), (3, 5'), (5', 2'), (2', 1)$; second cycle has the following four arcs $(2, 4), (4, 6'), (6, 4'), (4', 2)$; third cycle $(3, 5), (5, 7'), (7'5'), (5', 3)$ All the cycles are given in Figure 6.6. The number of aligned residues is 7 the total overlap works out to be 6.

6.7 Results

The entire procedure that calculates the overlap between contact maps of two proteins is implemented for the bench mark data set. The local alignment algorithm is chosen to be MSVNS as it is made publicly available by Pelta et al. [10] It should be noted that any fast algorithm can be plugged in as the local alignment algorithm in our approach. We consider all the pairs of proteins that are chosen by Pelta et al. [60] to facilitate comparison. The authors take a total of 161 pairs by choosing all pairs of proteins from within each fold except different chains of same proteins.

In the literature, the results are given in terms of total number of aligned residues and the total resultant overlap. Table 6.9 carries the result of the proposed algorithm for each fold. The paper [60] gives an overall result of overlap obtained to be 47,093 whereas the proposed algorithm obtained nearly 84% of the result by obtaining an overlap of 36,706.

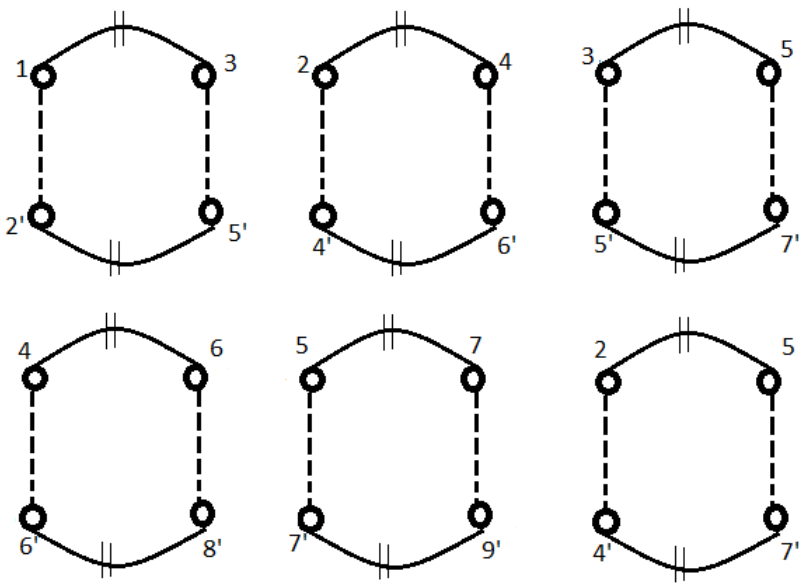


Figure 6.6: Representation of overlaps

These results are tabulated in Table 6.10. A further analysis is conducted into the results to understand the degradation of performance.

Table 6.9: Number of aligned values in Skolnick data set

Fold	No. of Aligned Residues	Maximum Overlap
Flavodoxin-like	5004	16120
Cupredoxin-like	3021	6102
Tim beta/alpha-barrel	8120	12140
Ferritin-like	550	2000
Microbial and ribonuclease	200	344

Table 6.10: Comparing results with MSVNS

Method	Protein Pairs	Total Aligned residues	Total overlap
Proposed	161	16895	36706
MSVNS [60]	161	15823	47093

6.7.1 Discussion

We analyzed our results and see that up to the Dynamic programming step of alignment of regions between proteins, our algorithm is on par with the heuristic algorithm. That is,

the regions that are obtained as aligned are also aligned by MSVNS. But the conflicts in alignment and removal of conflicting edges by MNCM algorithm, is leading to decrease in overlap eventually. For example, in Cupredoxin fold which has 11 proteins, due to many conflicts in the alignment of residues, the overlap obtained is 6102 as compared to 8150 by MSVNS. On the other hand, the approach is validated if we consider the fold of Tim beta/alpha-barrel fold which has 11 proteins. The contact maps of these proteins are seen to have very large clusters and the conflicts in regions is minimized apparently. The overall value of overlap obtained for this fold is 12140 which is better than 12127 that is obtained by MSVNS as reported in Table 6.11. Clearly the method is novel and works well in specific cases. When the protein contact maps have many small clusters, they are seen to lead to conflicts. Hence on lines of Bimal [19], we may have to apply a weighted non-crossing matching scheme which may help in retaining edges that will eventually maximize the overlap.

Table 6.11: Number of aligned and overlap values in Tim beta/alpha-barrel

Algorithm	Fold	No. of Aligned Residues	Maximum Overlap
Proposed	Tim beta/ alpha-barrel	**8120**	**12140**
MSVNS		8110	12127
Proposed	Cupredoxin-like	3021	6102
MSVNS		3231	8150

We now record the time taken for the whole exercise. Since each of the pairs of the smaller contact maps obtained from the dynamic programming approach can be submitted to MSVNS in parallel, it reduces the overall computation time in a considerable manner. We find that the preprocessing step takes about 30 minutes of time that includes extraction and matching all patterns extracted using *Approx_2D Pattern Matching* algorithm on Skolnick data set. The time taken by MSVNS to run on the entire data set is 5 hours 44 minutes, and the time taken on the smaller contact maps is almost 1/5-th time of about 1 hour 32 minutes as shown in the Table 6.12. If this step were parallelized, the time taken would have been almost negligible.

Table 6.12: Timing analysis for Skolnick dataset

MSVNS	Proposed Method
5hr. 44m.	2hr.

6.8 Conclusions

A novel approach to CMOP has been proposed using the two dimensional clusters of contact map. The general approach is to find matching clusters in both the contact maps which are pairs of small contact maps. These are submitted to an existing fast nearly optimal algorithm. The approach facilitates parallelization at this level since all the pairs of contact maps can be submitted to the algorithm in parallel. Then a merge algorithm is used in order to obtain the overall alignment. As a proof of concept we have chosen a tool called MSVNS for the nearly optimal alignment algorithm since it is publicly available for use. We obtain an overall result 47,093 overlapping edges whereas the proposed algorithm obtained nearly 84% of the result by obtaining an overlap of 36,706. On Tim beta/alpha-barrel fold, whose contact maps are seen to have very large patterns, our algorithm obtains a better overlap value of 12140 compared to 12127 of MSVNS. We can conclude that our method shows promise. In order to improve the merge procedure, following the ideas of the algorithm of Bimal [19], we may have to apply a weighted scheme which may help in retaining edges that will eventually maximize the overlap. The approach gives a parallelizable algorithm and hence should be of interest for online situations. The divide and conquer approach facilitates parallel implementation and hence the time taken for the alignment is effectively equal to the maximum time taken by the local alignments. Our implementations show that, on average, alignment on smaller contact maps takes less than one minute whereas existing alignment algorithms take more than 5 minutes between a pair of proteins.

Chapter 7

Conclusions and Future work

In this thesis we conduct a detailed analysis of contact maps in order to derive features that pertain to fold information. The main focus of this work is in proposing an alternate route to solve open problems of computational biology like protein secondary structure prediction, protein fold recognition, protein fold signatures and contact map overlap problem by mining contact maps. Somdatta et al. were the first to indicate that contact map analysis could be directly used for fold prediction problem [14]. They demonstrate in a few examples that proteins belonging to the same fold have similar contact maps.

Using the work in the literature that predicts contact maps from the primary amino acid sequence, we propose that using pattern features from contact maps would also be an ab-initio method. Hence feature extraction from the contact maps is the main stay of the thesis.

7.1 Conclusions

In chapter 3, we show how simple heuristics can extract statistics of secondary structure elements of helices and beta strands. The predictions have been tested using standard measures like Precision, Segment Overlap and on the predicted contact maps of the bench mark proteins the performance results are compared with those of the existing literature. Protein secondary structure prediction is achieved with an accuracy of 76% with a good accuracy obtained for helix prediction at 91% on par with the best results in the literature.

Further, we propose an *ExtractPatterns* algorithm which extracts clusters from the off-diagonal region in linear time. The number of non-trivial clusters are stored along with their statistics. A feature vector is constituted using both secondary structure related features viz. , number of helices, minimum helix length, maximum helix length, number

of beta sheets, minimum beta sheet and maximum beta sheet; as well as cluster features like number of clusters, minimum and maximum density as well directional features. We apply the standard data mining techniques to solve the protein fold recognition problem. Protein folds constitute a well-known unbalanced classification, a 27-class problem, which has been addressed using well-known one-against-one, one-against-others and all-versus-all methods using SVM's. There is just one work due to Krishnaraj et al. which shows an accuracy of 60.3% by using boosting approaches. We adopt a similar approach but with totally novel features extracted from the contact maps and use SMOTE based approach to balance the data. In chapter 4, we present this work on the imbalanced classification of protein fold recognition. SMOTE proves to be useful for the 27-way classification problem of protein fold prediction. Our emphasis is more on features derived from contact maps rather than on using state-of-art classifiers. Hence we do not experiment with highly improved versions of boosting algorithms that are available now [22] which may improve the results even further. With a standard model of combining SMOTE with C4.5 decision tree, the contact map features enhance the prediction accuracy to 65.25%. On the data set of predicted contact maps of All-Alpha Class, this approach resulted in an accuracy of 78% whereas the best in literature shows 78%. An additional advantage of our approach has been the reduced dimensionality of our feature vector which is 11 whereas literature uses more than 100 features on average.

We carry on the investigation to delve deeper in order to see if 2D-patterns have any correspondence with substructures within a fold. Are there specific patterns that correspond to a particular folding structure? This analysis turned out to be very fruitful. We use 2D-pattern matching algorithms to distinguish significant patterns.

We derive 'signature' rules for each protein fold using class based association rule mining algorithm. These rules give important insights into the significant 3D substructures that may be crucial for a protein to assume a specific fold conformation. Further, we demonstrate correspondence between the frequency of a 2D-pattern in a fold to the specific nature of substructure located within the tertiary structure and its functional importance. We validate these ideas further by using the presence or absence of these patterns as features and carry out the challenging fold recognition problem. Using class based association rule mining algorithm these signatures achieve the highest accuracy of 71.55% on the 27-way fold classification problem. On predicted contact maps of All-Alpha class, the prediction accuracy soared up to 85%. Alpha+Beta structural class poses challenge which needs to be investigated further.

In Chapter 6, we take up the challenge of the important NP-Hard problem in this area, namely the Contact map overlap problem. The 2D-pattern analysis carried so far gives

us a kind of a direct matching of regions between any two given protein contact maps. Firstly, we need to establish if these ideas do give a region-level matching and further, if this matching helps us in computing an alignment that maximizes the overlap between two proteins. This kind of investigation is new and hence promises a new way of approaching the well-studied CMO problem. We represent each protein as a set of the non-trivial clusters of contacts extracted from its contact map. We propose an approximate 2D-pattern matching algorithm using which a scoring matrix is constructed between all the pairs of clusters. Then use dynamic programming algorithm, to find a region level alignment.

Then we use an existing fast nearly optimal CMO algorithm to find alignment within the smaller contact maps. The results are merged in order to arrive at the overall alignment and thus the overlap is computed. This approach facilitates parallelization since all the pairs of regions can be submitted to the algorithm in parallel. Then a merge algorithm is used in order to obtain the overall alignment. As a proof of concept we have chosen a tool called MSVNS for the local alignment algorithm since it is made available for use. The result is not satisfactory since the average overlap obtained is 84% that of MSVNS. But there are best cases in which the algorithm exceeded the overlap obtained by MSVNS.

We can conclude that the method shows promise. It needs to be certainly further refined. The proposed algorithm is a parallel algorithm and hence reduces the computation time. Moreover, the alignment is in fact built based on sub-structure alignment and hence more structurally meaningful.

7.2 On Predicted Contact Maps

It is surprising for us that, through the stated prediction accuracy for contact maps is less than 30% [63] our algorithms seem to perform satisfactorily on predicted contact maps. We have shown that secondary structure prediction is successful with an accuracy of 76.8% on a sample test set. Further on the predicted contact maps of All-Alpha Class, the rule mining algorithm resulted in an accuracy of 85% whereas the best in literature shows 78%.

7.2.1 Sensitivity analysis

We show that our results are robust by doing the following experiment. We introduce 1% noise in the protein contact maps and implemented rule mining algorithm experiments. We find that the results do not suffer much due to the noise. We achieved highest accuracy of 85% on predicted contact maps of All-Alpha class.

7.2.2 Other contact maps

Current approaches construct protein contact maps using C_β atoms as nodes. We have also retrieved contact maps considering C_β atoms and we found that no significant variation is seen in the contact maps. It is a different issue altogether to consider interactions between side chains and not only the protein backbone. We do not know of any work that considers protein contact maps using side chain interactions. This issue needs to be further investigated.

7.3 Future directions

Clusters of contacts have been extracted using *Sig_Pattern* algorithm that are found to be significant for a fold. We need to investigate if, instead of using presence/absence of a pattern, the actual frequency of occurrence of pattern improves the performance of the classifier. Clearly there is immense scope for improvement to the solution for CMO. The merging algorithm is failing and we may have to intelligently merge so that necessary alignment edges are not removed. This is one direction which needs to be certainly investigated. Further, we feel that this solution can be better lifted up to the structure comparison in 3D directly. Our initial analysis shows that up to region level alignment, the proteins are being matched as 3D-structures and their mutual proximites very well. Hence instead of projecting these results onto one dimensions if we can lift them up to 3D we may gain better advantage and address the problem of structural alignment in known structures.

Zaki et al. initiated ideas in which the contact map clusters have been analyzed in order to predict the protein folding pathway, a grand challenge problem in this area. It is enticing to think that computational analysis of contact maps carried out in this thesis may help further these ideas!

Appendix A

The entire Data set given by Ding et al. containing proteins belonging to the four major structural classes in 27 protein folds is given in Table A.1.

Table A.1: The distribution of proteins among the 27-folds

Fold Name	Fold Index	Training Instances	Testing Instances
All-Alpha		**68**	**48**
Globin-like	1	11	8
Cytochrome -C	3	11	5
DNA-binding 3-helical bundle	4	18	10
4-helical up-and-down bundle	7	8	7
4-helical cytokines	9	12	10
EF-hand	11	8	8
All-Beta		**109**	**91**
Immunoglobulin-like β-sandwich	20	30	18
Cupredoxins	23	9	12
Viral coat and capsid proteins	26	16	13
ConA-like lectins/glucanases	30	7	6
SH3-like barrel	31	8	8
OB-fold	32	13	19
Trefoil	33	8	4
Trypsin-like serine proteases	35	9	4
Lipocalins	39	9	7
Alpha/Beta		**115**	**109**
(TIM)-barrel	46	29	24
FAD (also NAD)-binding motif	47	11	12
Flavodoxin-like	48	11	13
NAD(P)-binding Rossmann-fold	51	13	12
P-loop containing nucleotide	54	10	15
Thioredoxin-like	57	9	8
Ribonuclease H-like motif	59	10	14
Hydrolases	62	11	7
Periplasmic binding protein-like	69	11	4
Alpha+Beta		**38**	**62**
β grasp	72	13	8
Ferredoxin-like	87	13	27
Small inhibitors, toxins, lectins	110	12	27

The 27-way classification results obtained by the CBA classifier are given in Table A.2

Table A.2: 27- way classification

Class	Proposed Accuracy
1	85
3	100
4	96
7	60
9	78
11	100
20	100
23	65
26	100
30	75
31	63
32	100
33	70
35	0
39	70
46	100
47	93.3
48	81.2
51	92.8
54	73.6
57	66.6
59	80
62	40
69	76.9
72	0
87	80.7
110	69.2
Average	**71.55%**

References

[1] http://www.google.co.in.

[2] http://scop.mrc--.cam.ac.uk.

[3] http://distill.ucd.ie/distill.

[4] http://www.cs.waikato.ac.nz/ml/weka/.

[5] http://www.rcsb.org/pdb/home/home.do.

[6] http://www.pymol.org.

[7] http://www.molgen.mpg.de/~lappe/cmview/download.html.

[8] http://www.comp.nus.edu.sg/~dm2/.

[9] http://modo.ugr.es/es/book/export/s5/95.

[10] http://modo.ugr.es/jrgonzalez/msvns4maxcmo.

[11] P. K. Agarwal, Nabil H Mustafa, and Yusu Wang. Fast molecular shape matching using contact maps. *Journal of Computational Biology*, 14(2):131–143, 2007.

[12] R. Andonov, N. Yanev, and Noël Malod-Dognin. An efficient lagrangian relaxation for the contact map overlap problem. In *WABI 2008, 8th Workshop on Algorithms in Bioinformatics LNBI 5251, Springer–Verlag Berlin Heidelberg*, pages 162–173, 2008.

[13] C. B. Anfinsen. Studies on the principles that govern the folding of protein chains, 1972.

[14] P. Barah and S. Sinha. Analysis of protein folds using protein contact networks. *Pramana*, pages 369–78, 2008.

[15] D. Bashford, Cyrus Chothia, and Arthur M Lesk. Determinants of a protein fold: Unique features of the globin amino acid sequences. *J. of molecular biology*, 196(1):199–216, 1987.

[16] D. Bau, Alberto JM Martin, Catherine Mooney, Alessandro Vullo, Ian Walsh, and Gianluca Pollastri. Distill: a suite of web servers for the prediction of one, two and three-dimensional structural features of proteins. *BMC Bioinformatics*, 7(1):402, 2006.

[17] V. D. Blondel, Anahí Gajardo, Maureen Heymans, Pierre Senellart, and Paul Van Dooren. A measure of similarity between graph vertices: Applications to synonym extraction and web searching. *SIAM review*, 46(4):647–666, 2004.

[18] H. Breu and David G. Kirkpatrick. Unit disk graph recognition is np-hard. *Computational Geometry. Theory and Applications*, 9, 1993.

[19] J. J. Brijnesh and Klaus Obermayer. Bimal: Bipartite matching alignment for the contact map overlap problem. In *IJCNN*, pages 1394–1400, 2009.

[20] A. Caparara and G. Lancia. Structural alignment of large-size proteins via lagrangian relaxation. In *RECOMB annual international conference on computational molecular biology*, pages 100–108, 2002.

[21] N. V. Chawla. C4.5 and imbalanced data sets: investigating the effect of sampling method, probabilistic estimate, and decision tree structure. In *Proceedings of the ICML*, volume 3, 2003.

[22] N. V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. Smote: synthetic minority over-sampling technique. *arXiv preprint arXiv:1106.1813*, 2011.

[23] N. V. Chawla, A. Lazarevic, L. O. Hall, and K. W. Bowyer. Smoteboost: Improving prediction of the minority class in boosting. In *In proceedings of the seventh european conference on principles and practice of knowledge discovery in databass*, pages 107–119, Dubrovnik, Croatia, 2003.

[24] J. Cheng and Pierre Baldi. Improved residue contact prediction using support vector machines and a large feature set. *BMC Bioinformatics*, 8(1):113, 2007.

[25] W. Chmeilnicki and K. Stapor. An efficient multiclass support vector machine classifier for protein fold recognition. In *IWPACBB*, pages 77–84, 2010.

[26] P. Y. Chou and Gerald D Fasman. Empirical predictions of protein conformation. *Annual review of biochemistry*, 47(1):251–276, 1978.

[27] A. Dehzangi, Somnuk Phon Amnuaisuk, Keng Hoong Ng, and Ehsan Mohandesi. Protein fold prediction problem using ensemble of classifiers. In *Neural Information Processing*, pages 503–511, 2009.

[28] R. E. Dickerson. Cytochrome c and the evolution of energy metabolism. *Scientific american of fprints, 1464, publisher freeman*, 1980.

[29] C. H. Q. Ding and I. Dubchak. Multi-class protein fold recognition using support vector machines and neural networks. *Bioinformatics*, pages 349–358, 2001.

[30] C. Elkan. Boosting and naive bayesian learning. *Technical Report CS97-557 University of California,Sam Diego, CA*, pages –, 1997.

[31] W. Fan, S. J. Stolfo, J. Zhang, and P. K. Chan. Adacost:misclasification cost-sensitive boosting. In *In proceedings of sixth international conference on machine learning(ICML–99)*, pages 97–105, Slovenia, 1999.

[32] P. Fariselli and R Casadio. A neural network based predictor of residue contacts in proteins. *Protein Engineering*, 12(1):15–21, 1999.

[33] R. Fraser. A tale of two helices: A study of alpha helix pair conformations in three-dimensional space. *Master's thesis, Queen's University*, 2006.

[34] R. Fraser and J. Glasgow. A demonstration of clustering in protein contact maps for alpha helix pairs. In *ICANNGA*, pages 758–766, 2007.

[35] Y. Freund and R. E. Schapire. Experiments with a new boosting algorithm. In *In proceedings of the thirteenth international conference on machine learning, The mit press.*, pages 148–156, Morgan Kaufmann 1996.

[36] J. Garnier, D. J. Osguthorpe, and B Robson. Analysis of the accuracy and implications of simple methods for predicting the secondary structure of globular proteins. *J. of molecular biology*, 120(1):97–120, 1978.

[37] J. L. Gifford, P. Michael, W. Hans, and J. Vogel. Structures and metal–ion–binding properties of the ca2+ –binding helixloophelix ef–hand motifs. *Biochem. J. 405*, pages 199–221, 2007.

[38] U. Göbel, Chris Sander, Reinhard Schneider, and Alfonso Valencia. Correlated mutations and residue contacts in proteins. *Proteins: Structure, Function, and Bioinformatics*, 18(4):309–317, 1994.

[39] D. Goldman, Sorin Istrail, and Christos H Papadimitriou. Algorithmic aspects of protein structure similarity. In *Foundations of Computer Science, 1999. 40th Annual Symposium on*, pages 512–521, 1999.

[40] H. Guo and H. L. Viktor. Learning from imbalanced data sets with boosting and data generation: The databoost–im approach. In *SIGKDD explorations special issue on learning from imbalanced datasets*, pages 30–39, 2004.

[41] J. Han and M. Kamber. *Data mining: Concepts and techniques*. Elsevier, 2008.

[42] J. Hu, X. Shen, Y. Shao, C. Bystroff, and M. J. Zaki. Mining protein contact maps. In *2nd BIOKDD workshop on data mining in bioinformatics*, pages 196–204, 2002.

[43] S. Hua and Zhirong Sun. A novel method of protein secondary structure prediction with high segment overlap measure: support vector machine approach. *J. of molecular biology*, 308(2):397–407, 2001.

[44] N. Japkowicz and S. Stephen. The class imbalance problem: A systematic study. *Intelligent Data Analysis. J.*, 6(5):429–450, 2002.

[45] D. T. Jones, Daniel WA Buchan, Domenico Cozzetto, and Massimiliano Pontil. Psicov: precise structural contact prediction using sparse inverse covariance estimation on large multiple sequence alignments. *Bioinformatics*, 28(2):184–190, 2012.

[46] M. V. Joshi, V. Kumar, and R. C. Agarwal. Evalating boosting algorithms to classify rare classes: Comparison and improvements. In *In proceeding of the first IEEE international conference on data mining(ICDM'01)*, pages 983–990, 2001.

[47] G. Karypis. Better kernels and coding schemes lead to improvements in svm-based secondary structure prediction. Technical report, DTIC Document, 2005.

[48] H. Kim and Haesun Park. Protein secondary structure prediction based on an improved support vector machines approach. *Protein Engineering*, 16(8):553–560, 2003.

[49] J. M. Kleinberg. Authoritative sources in a hyperlinked environment. *J. of the ACM (JACM)*, 46(5):604–632, 1999.

[50] Y. Krishnaraj and Chandan K Reddy. Boosting methods for protein fold recognition: an empirical comparison. In *Bioinformatics and Biomedicine, 2008. BIBM'08. IEEE International Conference on*, pages 393–396, 2008.

[51] M. Kubat, Robert C Holte, and Stan Matwin. Machine learning for the detection of oil spills in satellite radar images. *Machine learning*, 30(2-3):195–215, 1998.

[52] G. Lancia, R. Carr, and S. Istrail. A branch and cut algorithm for the maximum contact map overlap problem. In *RECOMB annual international conference on computational molecular biology*, pages 193–202, 2001.

[53] Di. P. Lena, Ken Nagata, and Pierre Baldi. Deep architectures for protein contact map prediction. *Bioinformatics*, 28(19):2449–2457, 2012.

[54] B. Liu, W. Hsu, and Y. Ma. Integrating classification and association rule mining. *KDD '98*, pages 80–86, 1998.

[55] F. Malucelli and Daniele Pretolani. Efficient labelling algorithms for the maximum noncrossing matching problem. In *Combinatorial Optimization*, pages 299–301. 1992.

[56] C. Mooney and Gianluca Pollastri. Beyond the twilight zone: automated prediction of structural properties of proteins by recursive neural networks and remote homology information. *Proteins: Structure, Function, and Bioinformatics*, 77(1):181–190, 2009.

[57] F. Morcos, Andrea Pagnani, Bryan Lunt, Arianna Bertolino, Debora S Marks, Chris Sander, Riccardo Zecchina, José N Onuchic, Terence Hwa, and Martin Weigt. Direct-coupling analysis of residue coevolution captures native contacts across many protein families. *Proceedings of the National Academy of Sciences*, 108(49):E1293–E1301, 2011.

[58] O. Olmea and Alfonso Valencia. Improving contact predictions by the combination of correlated mutations and other sources of sequence information. *Folding and Design*, 2:S25–S32, 1997.

[59] A. R. Ortiz, Andrzej Kolinski, Piotr Rotkiewicz, Bartosz Ilkowski, and Jeffrey Skolnick. Ab initio folding of proteins using restraints derived from evolutionary information. *Proteins: Structure, Function, and Bioinformatics*, 37(S3):177–185, 1999.

[60] D. A. Pelta, R. Gonzalez Juan, and Marcos Moreno Vega. A simple and fast heuristic for protein structure comparison. *BMC Bioinformatics*, pages 1–16, 2008.

[61] Di. Lena. Pietro, Piero. Feriselli, Luciano Margara, Macro Vassura, and Rita Casadio. Fast overlapping of protein contact maps by alignment of eigenvectors. *Bioinformatics*, pages 2250–2258, 2010.

[62] M. Punta and Burkhard Rost. Profcon: novel prediction of long-range contacts. *Bioinformatics*, 21(13):2960–2968, 2005.

[63] B. Rost and C. Sander. Prediction of protein secondary structure at better than 70% accuracy. *J. Mol. Biol.,*, pages 584–599, 1993.

[64] B. Rost, C. Sander, and Reinhard Schneider. Redefining the goals of protein secondary structure prediction. *J. of molecular biology*, 235(1):13–26, 1994.

[65] R. E. Schapire. Explaining adaboost. In *Empirical Inference*, pages 37–52. 2013.

[66] R. E. Schapire and Y. Singer. Improved boosting algorithms using confidence–rated predictions. *Machine learning*, pages 297–336, 1999.

[67] H. Schwenk and Y. Bengio. Boosting neural networks. *Neural computation*, pages 1869–1887, 2000.

[68] K. Segla, Philippe Galinier, and Giuliano Antoniol. Enhancing a tabu algorithm for approximate graph matching by using similarity measures. In *EvoCOP*, pages 119–130, 2010.

[69] M. T. A. Shamim, Anwaruddin, and H. A. M. Nagarajaram. Support vector machine–based classification of protein folds using the structural properties of amino acid residues and amino acid residue pairs. *Bioinformatics*, pages 3320–3327, 2007.

[70] A. Sharma, Kuldip K Paliwal, Abdollah Dehzangi, James Lyons, Seiya Imoto, and Satoru Miyano. A strategy to select suitable physicochemical attributes of amino acids for protein fold recognition. *BMC bioinformatics*, 14(1):233, 2013.

[71] J-Y. Shi and Y-N. Zhang. Fast scop classification of structural class and fold using secondary structure mining in distance matrix. In *PRIB2009, LNBI*, pages 344–353, 2009.

[72] Y. Shibberu and A. Holder. A spectral approach to protein structure alignment. *J. IEEE/ACM Transactions on Computational Biology and Bioinformatics*, pages 867–875, 2011.

[73] Y. Shibberu, A. Holder, and K. Lutz. Fast protein structure alignment. In *LNCS, Springer–Verlag Berlin Heidelberg*, pages 152–165, 2010.

[74] V. A. Simossis and Jaap Heringa. Local structure prediction of proteins. In *Computational Methods for Protein Structure Prediction and Modeling*, pages 207–254. 2007.

[75] D. M. Strickland, E. Barnes, and J. S. Sokol. Optimal protein structure alignment using maximum cliques. *J. Operations research*, pages 389–402, 2005.

[76] A. N. Tegge, Zheng Wang, Jesse Eickholt, and Jianlin Cheng. Nncon: improved protein contact map prediction using 2d-recursive neural networks. *Nucleic Acids Research*, 37(suppl2):515–518, 2009.

[77] K. M. Ting. A comparative study of cost-sensitive boosting algorithms. In *In proceedings of the 17th international conference on machine learning*, pages 983–990, Stanford University, CA, 2000.

[78] C. Travaglini, S. Gianni, and V. Morea. Exploring the cytochrome c folding mechanism: cytochrome c552 from thermus thermophilus folds through an on–pathway intermediate. *J. Biol Chem*, pages 41136–41140, 2003.

[79] M. Tress, T. H. C. Hsien, and G. Wang, G. López. Domain definition and target classification for casp6. In *Proteins*, 2005.

[80] M. Turcotte, H. S. Muggleton, and M. J. E. Sternberg. Automated discovery of structural signatures of protein fold and function. *J. Mol. Biol*, pages 591–605, 2001.

[81] M. Vassura, Luciano Margara, Pietro Di Lena, Filippo Medri, Piero Fariselli, and Rita Casadio. Reconstruction of 3D structures from protein contact maps. *Computational Biology and Bioinformatics, IEEE/ACM Transactions on*, 5(3):357–367, 2008.

[82] M. Vendruscolo and Eytan Domany. Pairwise contact potentials are unsuitable for protein folding. *The J. of Chemical Physics*, 109(24):11101–11108, 1998.

[83] M. Vendruscolo, Edo Kussell, and Eytan Domany. Recovery of protein structure from contact maps. *Folding and Design*, 2(5):295–306, 1997.

[84] M. Vendruscolo, B. Subramanian, I. Kanter, E. Domany, and J. Lebowitz. Statistical properties of contact maps. *Physical Review E*, pages 977–84, 1999.

[85] M. Vendruscolo, Balakrishna Subramanian, Ido Kanter, Eytan Domany, and Joel Lebowitz. Statistical properties of contact maps. *Phys. Rev. E*, 59:977–984, 1999.

[86] Saraswathi Vishveshwara, KV Brinda, and N Kannan. Protein structure: insights from graph theory. *Journal of Theoretical and Computational Chemistry*, 1(01):187–211, 2002.

[87] A. Vullo, Ian Walsh, and Gianluca Pollastri. A two-stage approach for improved prediction of residue contact maps. *BMC Bioinformatics*, 7(1):180, 2006.

[88] Allen J W, Daltrop O, Stevens J, and Ferguson S J. Ctype cytochromes: diverse structures and biogenesis systems pose evolutionary problems. *Philos. Trans. R. Soc. London*, pages 255–266, 2008.

[89] J. Wang, Mantao Xu, Hui Wang, and Jiwu Zhang. Classification of imbalanced data by using the smote algorithm and locally linear embedding. In *Signal Processing, 2006 8th International Conference on*, volume 3, pages –, 2006.

[90] K. J. Won, Thomas Hamelryck, Adam Prügel-Bennett, and Anders Krogh. An evolutionary method for learning hmm structure: prediction of protein secondary structure. *BMC bioinformatics*, 8(1):357, 2007.

[91] S. Wu, Andras Szilagyi, and Yang Zhang. Improving protein structure prediction using multiple sequence-based contact predictions. *Structure*, 19(8):1182–1191, 2011.

[92] S. Wu and Yang Zhang. A comprehensive assessment of sequence-based and template-based methods for protein contact prediction. *Bioinformatics*, 24(7):924–931, 2008.

[93] W. Xie and N. V. Sahinidis. A reduction-based exact algorithm for the contact map overlap problem. *J. Computaional Biology*, pages 637–654, 2007.

[94] J. Xu, F. Jio, and B. Berger. A parameterized algorithm for protein structure alignment. *J. Computaional Biology*, pages 564–577, 2007.

[95] L. A. Zager and George C Verghese. Graph similarity scoring and matching. *Applied mathematics letters*, 21(1):86–94, 2008.

[96] M. J. Zaki, J. Hu, and C. Bystroff. Methods for mining contact maps. *Data mining: Next generation challenges and future directions, AAAI/MIT Press*, pages 291–314, 2004.

[97] M. J. Zaki, Vinay Nadimpally, Deb Bardhan, and Chris Bystroff. Predicting protein folding pathways. In *ISMB/ECCB (Supplement of Bioinformatics)*, pages 386–393, 2004.

[98] A. Zemla, Česlovas Venclovas, Krzysztof Fidelis, and Burkhard Rost. A modified definition of sov, a segment-based measure for protein secondary structure prediction assessment. *Proteins: Structure, Function, and Bioinformatics*, 34(2):220–223, 1999.

[99] Y. Zhao and G. Karypis. Prediction of contact maps using support vector machines. *International J. on artificial intelligence tools*, pages 849–866, 2005.

[100] H. Zhou and Jeffrey Skolnick. Protein model quality assessment prediction by combining fragment comparisons and a consensus cα contact potential. *PROTEINS: Structure, Function, and Bioinformatics*, 71(3):1211–1218, 2008.