Dissertation

Large Margin Methods for Partner Specific Prediction of Interfaces in Protein Complexes

Submitted by

Fayyaz ul Amir Afsar Minhas

Department of Computer Science

In partial fulfillment of the requirements

For the Degree of Doctor of Philosophy

Colorado State University

Fort Collins, Colorado

Spring 2014

Doctoral Committee:

Advisor: Asa Ben-Hur

Bruce Draper
Charles Anderson
Christopher Snow

Abstract

Large Margin Methods for Partner Specific Prediction of Interfaces in Protein Complexes

The study of protein interfaces and binding sites is a very important domain of research in bioinformatics. Information about the interfaces between proteins can be used not only in understanding protein function but can also be directly employed in drug design and protein engineering. However, the experimental determination of protein interfaces is cumbersome, expensive and not possible in some cases with today's technology. As a consequence, the computational prediction of protein interfaces from sequence and structure has emerged as a very active research area. A number of machine learning based techniques have been proposed for the solution to this problem. However, the prediction accuracy of most such schemes is very low.

In this dissertation we present large-margin classification approaches that have been designed to directly model different aspects of protein complex formation as well as the characteristics of available data. Most existing machine learning techniques for this task are partner-independent in nature, i.e., they ignore the fact that the binding propensity of a protein to bind to another protein is dependent upon characteristics of residues in both proteins. We have developed a pairwise support vector machine classifier called PAIRpred to predict protein interfaces in a partner-specific fashion. Due to its more detailed model of the problem, PAIRpred offers state of the art accuracy in predicting both binding sites at the protein level as well as inter-protein residue contacts at the complex level. PAIRpred uses sequence and structure conservation, local structural similarity and surface geometry, residue solvent exposure and template based features derived from the *unbound* structures

of proteins forming a protein complex. We have investigated the impact of explicitly modeling the inter-dependencies between residues that are imposed by the overall structure of a protein during the formation of a protein complex through transductive and semi-supervised learning models. We also present a novel multiple instance learning scheme called MI-1 that explicitly models imprecision in sequence-level annotations of binding sites in proteins that bind calmodulin to achieve state of the art prediction accuracy for this task.

Acknowledgements

I am very grateful to my research adviser, Dr. Asa Ben-Hur for the time, insight, motivation, direction and kindness he rendered to me. During the three years I worked with him as a Ph.D. student, we exchanged more than 2,000 emails which corresponds, on average, to about two emails per day. I do not have similar statistics on the time that we spent during meetings, brain-storming sessions, presentations and discussions. However, I was always much more relaxed, focused and motivated after coming back from a meeting with him. I do hope to continue to have him as a mentor and friend in the future as well.

I also thank my doctoral dissertation committee: Dr. Charles Anderson, Dr. Bruce Draper and Dr. Christopher Snow for their time, fruitful suggestions and guidance. My field of study is inter-disciplinary and I did not have much of a background in protein biochemistry. I would especially like to thank Dr. Jeffrey Hansen and Dr. Brian Geiss for their time, genorosity and patience in helping me remedy that. I also owe thanks to the magnificent and inspiring professors and researchers at CSU like Dr. Sanjay Rajopadhye, Dr. Adele Howe, Dr. Robert France, Dr. A.P. Willem Bohm, Dr. Darrel Whitley, Dr. Corey Broeckling, Dr. James Cooney and Dr. Bernard Rollin. I would also like to thank Sarah McCormick and Aislinn O'Callahan at the Institute of International Education, Christy Eylar and Jenn Christ at CSU office of International Programs, Amer Malik, Nadia Kamal and Nadia Omar at United States Education Foundation in Pakistan, Dr. Abdul Jalil, Dr. Mutawarra Hussain, Dr. Javaid Khurshid and Muhammad Fayyaz at Pakistan Institute of Engineering and Applied Sciences for their administrative help and guidance.

I would like to express my gratitude towards my family, especially my mother and my four year old son Haider. Both of you are the reason for me to keep going. Haider is also

responsible for choosing the colors for some of the graphics in this dissertation. There is an Arabic saying about friends which roughly translates to *"Every person has siblings from other parents"*. I am lucky to have such friends. I am thankful to Ali and Maria Kamal, Nasir, Ani, Syed Ata, Bob and Liz Hand, Bahauddin, Anna, Troy and Jamie Fine, Omar, Bashir, Dieudo, Martin, Sofia, Ayesha, Monica, Fathallah, Younis, Saad, Mumtaz Hussain, Dr. Masroor Kakakhel, Ibrahim and Eva for keeping me sane through the four years of my Ph.D. In particular, I would like to thank Faiq Majeed for his support. I must also extend my thanks to the members of the bioinformatics group at CSU: Mike, Mark, Indika, Fahad, Kiley and Artem. If I missed your name in this list and you think it belongs here, I apologize.

Great thanks are also due to Leif Anderson for building this LaTeX document class and Fran Campana in helping me meet the graduate school formatting requirements.

Lastly, I would like to thank the Fulbright scholarship program of the U.S Department of State and the Higher Education Commission of Pakistan for their funding for my Ph.D. studies. My gratitude for this scholarship goes way beyond my thanks for the university fees, monthly stipends, learning resource allowances, travel fares and conference fundings that it provided for this research to be possible. My primary reason to thank this scholarship is the fact that it provided me with an opportunity to expand my horizons and play my role in bringing *a little more knowledge, a little more reason, and a little more compassion into world affairs and thereby increase the chance that nations will learn at last to live in peace and friendship.*

Table of Contents

CHAPTER 1

Introduction

Even though our scientific understanding of life, its origins and its mechanisms is still far from complete, one of the biggest strides made by man in this quest has been to link biology with chemistry and physics. We have been constantly trying to understand the chemical processes and physical phenomenon that make life possible. The discovery of the cell itself (1665 A.D.), followed by that of the Deoxyribonucleic acid (DNA) (1869 A.D.) and the establishment of modern protein chemistry (ca. 1820 A.D.) have been the major milestones in this multidisciplinary view of biology. We now know that DNA is the molecule of life. It links all biological life in a four-lettered code and is responsible for all the diversity, beauty and vibrancy that we observe as life. However, the mechanisms through which DNA does that involve other molecules such as Ribonucleic acid (RNA), proteins, carbohydrates, lipids, vitamins etc. According to the central dogma of molecular biology [1], DNA is used as a template to build proteins which then form the functional backbone of innumerable biologically important processes.

The role proteins play in cellular functions can be appreciated by considering that approximately 50% of the dry weight of the human body is protein [2]. Another example that illustrates the importance of proteins is that of human blood which consists of red and white blood cells. A single protein called hemoglobin constitutes about 92% of the dry weight of all red blood cells [3]. Hemoglobin gives the blood its distinct red color and is involved in the transfer of oxygen to cells which is absolutely critical for survival. Other functions of

proteins include but are not limited to: enzymatic processing, cell signaling, transport, immunological responses, muscular contractions, hormonal processes, structural stability, and gene expression control [1].

As a consequence of the functional importance of proteins, understanding how proteins work, lies at the heart of comprehending the mechanics of cellular function. Apart from their biological importance, proteins are also very important in medicine and industries such as textile and agriculture. Understanding protein function is crucial for understanding disease mechanisms and developing drugs. This is illustrated by the fact that more than 80% of current pharmaceutical targets are proteins [1].

The functional diversity of proteins stems from their ability to interact with other macromolecules and ligands. Of particular interest are the interactions of proteins with other proteins. Such interactions lie at the core of a huge number of biological processes and disease mechanisms. For example, hemoglobin is a complex of multiple protein chains and it is only through the formation of this complex that oxygen can be carried to cells. A single mutation in the gene that is responsible for creating hemoglobin changes how its individual chains interact with one another, resulting in Sickle-cell disease [1].

When proteins interact with one another to form protein complexes, components of the individual proteins physiochemically bind to one another at locations called *interaction* or *binding sites*. This study is aimed at the development of computational methods for predicting binding sites in protein-protein interactions from data about the proteins forming the complex. Knowledge about the binding sites of a protein can help in identifying its function, understanding the underlying biochemistry of different diseases and biological processes and also in drug development.

Location of the binding sites in a protein complex can be obtained experimentally by determining the structure of the complex using X-ray crystallography, Nuclear Magnetic Resonance (NMR) spectroscopy, or electron microscopy [1]. Alternatives to these structure determination schemes include biochemical assays such as repetitive yeast two-hybrid techniques [4] or mutagenesis experiments [5] which, essentially, search for locations in a protein where a mutation or change will make the formation of the complex impossible. However, present day structure determination technologies cannot determine the structure of every protein and are tedious and expensive to perform. For example, a study aimed at obtaining protein crystals for X-ray crystallography of all the 1,877 protein targets in the bacterium *T. maritima* resulted in a success rate of only 24% [6]. Co-crystallization, i.e., the simultaneous crystallization of bound proteins in a protein complexes is even more difficult. The average cost of successfully determining the structure of a protein excluding capital expenditures is roughly $150,000 [7], with about 60% of the cost being a consequence of failed attempts at crystallization [8]. Biochemical assays suffer from similar problems. As a consequence, computational prediction of binding sites has emerged as an important area of research. Predictions generated by computational techniques can not only be directly used in studying underlying complex biology, but can also be used to direct experimental studies.

1.1. Research objectives and thesis organization

Prediction of protein interfaces is a difficult problem owing to a variety of challenges. Our first step is to identify these challenges, as well as the requirements and expectations form a protein interface predictor (Chapter 2). We have chosen to model the partner-specific nature of protein binding as a large-margin classification problem. As explained in Chapter 2, most existing research in this domain does not model this aspect of the problem. Our prediction

model, presented in Chapter 3, shows that modeling the partner-specific nature of protein binding can not only provide better accuracy but it can also allow *in-silico* analyses which are beyond the power of partner-independent predictors. One of the research objectives of this work is to identify what kind of features are useful for an interface predictor. We investigate both sequence and structure based features in this work. Specifically, we have analyzed the efficacy of features derived from protein sequence alone and in conjunction with structure based features such as residue exposure and local surface geometry (Chapter 3). We have also probed the usefulness of local structural alignments, structural conservation, local geometric similarity and template based features for interface prediction (Chapter 4). Most machine learning based protein binding prediction schemes do not consider the inter-dependencies or labeling constraints between residues from the same protein or residue pairs from the same complex when generating predictions. One of the research goals of this work was to develop transductive and semi-supervised prediction schemes which explicitly consider such inter-dependencies (Chapter 5). Another research goal was to develop a prediction scheme that can model imprecision in binding site annotations or labels for protein sequences that bind Calmodulin. The resulting model, based on the multiple instance learning framework, is presented in Chapter 6. The conclusions from this research along with recommendations and directions for future work are discussed in Chapter 7.

We start off with a brief introduction to proteins and protein interfaces in order to assist the reader in understanding the rest of the thesis.

1.2. Characteristics of proteins

In this section, we present an overview of concepts related to proteins that are relevant for understanding the binding or interactions of proteins.

1.2.1. Sequence and structure. Proteins, like other important biological macromolecules, are biopolymers, i.e., they are composed of smaller subunits. In proteins, these subunits are called amino acids. The number of amino acids found in nature is 20. A protein can be thought of as an arbitrary-length sequence or chain of these amino acids. The median length of human proteins is 375 amino acids. Due to the presence of physiochemical interactions among different amino acid residues in the chain, the chain *folds* into a three dimensional structure. All the information required by a protein to fold into its three dimensional structure is encoded in the sequence of its amino acids [9]. The structure of a protein is determined by a complex interplay of covalent and non-covalent interactions between its amino acids, the environment in which the protein lies, its binding partners, presence of water molecules and a number of other factors.

The structure of a protein can be viewed at four different levels: primary, secondary, tertiary and quaternary (see Figure 1.1). Primary structure is the sequence of amino acids of the protein. The amino acids are held together by peptide bonds to form the backbone structure of the protein, and as a consequence a protein is sometimes called a polypeptide. Backbone atoms from neighboring amino acids in the three dimensional structure of the protein can interact with one another through hydrogen bonds and give rise to secondary structures such as α-helices, β-sheets, and loops. Secondary structures in a protein chain arrange themselves in a configuration called the tertiary structure of the protein. Folding of the protein into its tertiary structure allows residues that are not neighbors in sequence to come in spatial proximity, and this fact has important consequences for protein function and interactions. The 'branches' coming out of the backbone structure of the protein in Figure 1.1 are the *side chains* corresponding to different residues. It is the atomic composition of these

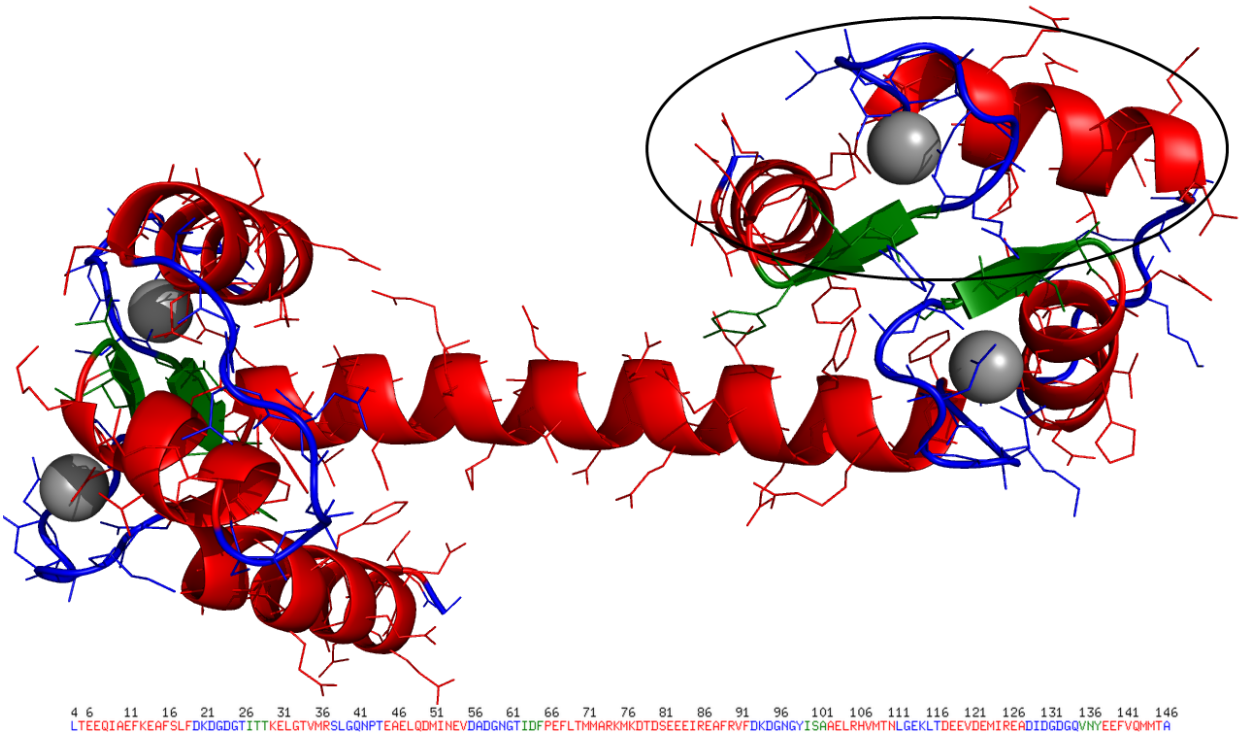


FIGURE 1.1. Primary, secondary and tertiary structure of the protein calmodulin (PDB ID: 1CLL [10]). The primary structure (amino acid sequence) is shown at the bottom of the figure. α-helices, β-sheets and loops are shown in red, green and blue, respectively. The ellipse encloses one of the four EF-Hand calcium binding motifs in calmodulin. Calcium ions bound by the protein are shown as gray spheres. Image taken from the Protein Data Bank (PDB) [11]. For more information, the interested reader is referred to [12].

side chains which gives different physiochemical properties to different amino acids. The backbone structure of all amino acids is identical. Proteins are composed of domains which are units in the protein that can evolve, function and exist independently of the rest of the protein chain. The protein shown in Figure 1.1 is the calcium sensing protein calmodulin and it contains a single domain called the EF-hand domain. This domain is also found in other calcium binding proteins such as LCP1. Proteins that contain the same domain are said to belong to the same family.

An important concept related to protein structure is that of structural motifs. Structural motifs are structurally similar neighborhoods that can occur in many proteins. Unlike domains, structural motifs cannot exist independently. Calmodulin has four EF-hand calcium binding structural motifs shown in Figure 1.1. A related concept is that of sequence motifs which are biologically significant amino acid patterns (for proteins) or nucleotide patterns (for DNA and RNA). For example, the consensus sequence of the EF-hand calcium binding structural motif is ExxxxxxxxDx[DN]x[SDN]Gx[LVI]x[ESD]xxE where 'x' denotes a don't-care position and residues in square brackets can substitute one another at the same position. Multiple protein chains interact with one another to give rise to the quaternary structure of a protein also called a protein complex. For the quaternary structure of calmodulin in complex with a small peptide, see Figure 1.2.

1.2.2. Function and evolution of proteins. The structure of a protein determines its function. For example, the ability of calmodulin to bind calcium stems from calmodulin having 4 calcium binding EF-hand structural motifs. The sequence-structure-function relationship lies at the core of structural biology.

Proteins are related to one another through evolution. Evolutionary relationships between proteins can be established by calculating the degree of sequence or structural similarity between proteins. Proteins that have shared evolutionary ancestry are called homologous proteins. Protein structure has been observed to be more conserved than both the sequence and function of proteins. Proteins with different sequences can fold into the same tertiary structure and perform very similar functions. The surface of a protein is less conserved than its core as many conserved residues are buried in the core to confer stability to the protein. A strong correlation has been observed between the fractional burial of a residue and its

rate of evolution [13]. Residues involved in binding are under functional constraints, and as a consequence, they evolve more slowly than other surface residues [14]. Such evolutionary relationships between proteins can help in identifying binding sites, as closely related proteins are likely to exhibit similar interaction patterns. However, the degree of conservation needs to be very high to infer that two similar proteins would bind to similar targets [15].

1.2.3. Motion. Another interesting behavior of proteins is their motion. Proteins exist in equilibrium between many different conformations, called sub-states and they continuously move from one state to another (see Figure 1.2 for an illustration of this concept). Crystallographic structures usually capture the most stable conformation with the lowest energy. These conformations can also be affected by temperature, solvent properties, and binding to other ions or molecules. Structural dynamics of a protein in its native, i.e., folded, structure has implications on the interaction behavior of the protein.

1.3. Formation of protein complexes

Thermodynamically, the formation of protein complexes can be explained by the reduction in free energy resulting from action of non-covalent interactions between the participating proteins. An example of this phenomenon is the burial of non-polar residues as a consequence of the hydrophobic effect which entails a decrease in enthalpy and an increase in the entropy of the water molecules surrounding the individual proteins before complex formation. The strength of binding, also known as binding affinity, is measured by the free energy of binding which is the difference between the free energy of the complex and the sum of the free energies of the unbound components. This energy is usually very small (-2.5 to -22 kcal/mol, the negative sign indicating favorable energy change) and as a consequence,

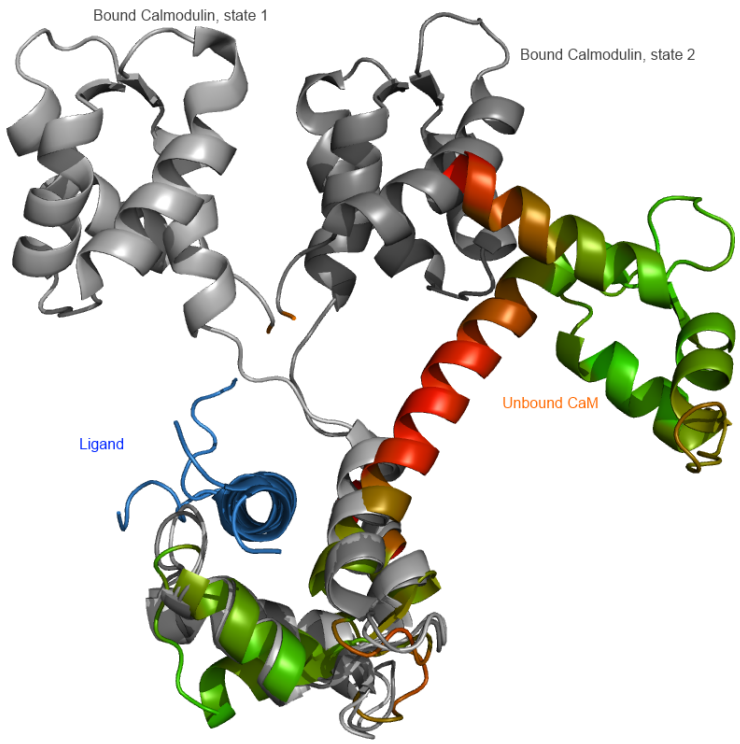


FIGURE 1.2. Motion and binding associated conformational change in proteins. Shown here, in shades of gray, are two states in the NMR structure (PDB ID: 2L53) of calmodulin in complex with a small polypeptide ligand (blue). This gives an idea of the extent of variation between different structural states of Calmodulin. Also shown is the aligned structure of calmodulin (PDB ID: 1CLL) colored (green to red) with respect to the b-factor of different residues in the protein when it is not bound to the ligand. B-factor is a measure of the deviation of atoms from their average locations (captured in the X-ray crystal structure) as a consequence of kinetic vibrations or disorder. Note that the unbound conformation is significantly different from the bound conformations with changes not only in the tertiary structure but also in secondary structure components.

protein complexes are only marginally stable [1]. For comparison, consider that breaking a single covalent bond requires 65-175 kcal/mol.

Binding of proteins to other proteins involves the same interactions as in protein folding and the only significant difference between inter-protein and intra-protein interactions is the lack of chain connectivity in the former. Typically, some degree of change in the conformation

or structure of the individual proteins is also associated with complex formation. Figure 1.2 shows the conformation change in Calmodulin when it binds a small protein.

In order to explain the mechanics of molecular recognition in proteins with respect to their structural and physiochemical properties, a number of models have been proposed. A thorough understanding of these models is crucial, as they not only explain how the process of binding takes place, but also capture different aspects of complex formation that are essential for the effective computational modeling of the problem of predicting binding sites in protein complexes.

One of the first such models is the *lock-and-key* model [16] which states that shape complementarity between the protein and the binding molecule is essential for binding to occur. However, this model does not fully describe protein binding, as shape complementarity, although playing an important role, is not the sole basis for binding [17].

A more realistic view is provided by the *induced-fit* model which states that together with shape complementarity, the binding process is also driven by non-covalent intermolecular forces such as van der Waals interactions, hydrophobic effects and hydrogen bonding. It also states that the binding process can cause conformational changes in the protein, leading to an induced fit of the binding partner to the protein [18]. Thus, along with shape complementarity, complementarity in the physiochemical properties of the two proteins determines binding. For example, hydrogen bond donors in a protein occur opposite to hydrogen bond acceptors in its binding partner, non-polar groups occur opposite to other non-polar groups and positive charges occur opposite to negative charges and so on.

A more detailed model for binding is the *conformational selection* model discussed in [19]. According to this model, a protein molecule exists in equilibrium between different conformations and this equilibrium is shifted to the conformation that exhibits the lowest energy state for the formation of the protein complex with another protein.

Based on these models, the task of predicting binding sites in proteins translates to *identifying areas in a protein where complementarity in shape and physiochemical properties between the protein and its binding partner either pre-exists or can result from a binding-triggered conformational change.*

1.4. Defining binding sites and interfaces

In the context of protein interactions, we propose to discriminate between a *binding site* and the *interface* in a complex as follows: the region on a protein that is involved in an interaction with another protein is called its binding site, whereas the group of interacting residues in a complex is called the *interface* of the complex. Note that, given the interface of a complex, it is trivial to determine the binding region on each protein in the complex.

Given the structure of the protein complex, we can identify two residues as interacting if their minimum inter-atomic distance is less than a certain threshold (say, 6.0 Å) [12, 20]. Two residues within this distance are not guaranteed to interact, but it is one of the easiest ways of defining protein interfaces and has been used by a large number of researchers in the field [20]. Creation of the protein complex entails burial of previously surface exposed residues of the individual proteins. The degree of surface exposure of a residue, called its accessible surface area (ASA), can be measured from the structure of a protein. Another method of finding which residues constitute the binding site for an interaction is to calculate the change in the ASA of residues upon complex formation.

1.5. Categorization of protein complexes

In this section we discuss different categories of protein complexes [21] in order to narrow down our specific area of study. A protein chain can form a complex with other copies of the same protein chain. Such complexes are called homo-dimers, homo-trimers, homo-tetramers and so on, depending upon the number of repetitions of the chain in the complex. Protein complexes composed of different protein chains are called hetero-complexes. In this thesis, our focus is the interfaces of hetero-complexes only.

Protein complexes can either be obligate or non-obligate [22]. If the proteins in a complex can exist as independent tertiary structures then such a complex is called non-obligate. Non-obligate complexes can either be transient or permanent depending upon whether the complex breaks down after formation *in vivo* or not. Transient complexes are more difficult to predict than permanent complexes. This is because proteins forming permanent complexes exhibit more evolutionary conservation, co-evolution[1], co-expression[2] and co-localization[3] in the cell and lesser degree of conformation change. Also, the amount of buried surface area and the amount of decrease in free energy for permanent and obligate complexes is larger than that for transient interactions [23]. In this study, we focus only on non-obligate transient complexes. Another categorization of protein complexes springs from their biological or functional context. Based on this, protein complexes can be divided into a large number of groups such as enzyme-substrate, antibody-antigen, receptor-hormone, etc. In this work, we focus on protein complexes in general, regardless of their function.

[1]Co-evolution is the phenomenon in which a change in one protein triggers a change, through evolution, in another protein.
[2]Two proteins are said to be co-expressed when they appear at the same time from DNA.
[3]Two proteins are said to be co-localized when they appear in the same cellular component.

1.6. Properties of protein interfaces

Some of the general features that distinguish binding sites from other regions include structure, physiochemical properties, evolutionary conservation, etc. In this section, similarities and differences between the characteristics of binding vs. non-binding regions are discussed. Information contained in this section is expected to help explain the choice of features made in different computational binding site prediction methods presented later. However, it should also be noted that different proteins can have very different binding site characteristics.

1.6.1. Sequence properties. The composition and propensity of different residues has been observed to be different in binding areas in comparison to non-binding areas. Such differences were employed as features in [24] for protein-protein interaction site prediction. For example, the propensity of Tryptophan to occur in protein-protein interfaces was found to be higher than that of Alanine [21]. The characterization of these differences using a variety of feature representations allows for sequence based prediction of protein binding sites.

1.6.2. Structural properties. Interfaces between proteins and smaller substrates (e.g., caffeine) are typically cavities and concave clefts. The surface area of proteins buried in these types of interactions is usually smaller (up to a few hundred $\mathring{A}^2$) in comparison to the flatter, much larger (700-2000 $\mathring{A}^2$), and more structurally intricate surfaces involved in protein-protein interactions [25]. Protein interfaces usually exhibit imperfect shape complementarity which optimizes non-covalent interactions, especially van der Waals forces, between the partners. Atoms in protein interfaces are also more tightly packed in comparison to non-binding surface components.

It should be noted that residues forming a binding site in a protein need not be contiguous in sequence. This holds for most types of protein interactions [12, 26]. In terms of secondary structure, binding areas on proteins tend to favor β-sheets in comparison to α-helices. Moreover, loops in binding sites tend to be longer [27].

Most of the time, the residues involved in forming an interaction lie on the surface of a protein, which suggests the use of structural properties such as solvent accessible surface area (ASA) [28, 29], protrusion index and depth index [30] for prediction of binding sites. However, at times, the binding process itself can result in a conformational change in the protein leading to changes in the degree of solvent exposure of different residues in the protein [20]. Features based on these conformational changes have been utilized in [30] for the prediction of hotspot residues (details in section 1.7).

Some of the structural descriptors used in computational methods for binding site prediction are [20]: neighbor list (residues that lie in spatial proximity to the residue in question), ASA (can be calculated from protein structure using programs like DSSP [31]), relative ASA (ASA expressed as a fraction of the overall surface of the residues that is exposed to the solvent), residue interface propensity [32], B-factor (approximates the flexibility of a residue in the crystal structure), secondary structure, etc. The challenge in using unbound structures for training a binding predictor is that proteins can undergo significant structural changes upon binding.

1.6.3. Physiochemical properties. In terms of their physiochemical properties, protein-protein interaction sites are often less polar and more hydrophobic in comparison to the rest of the protein [33]. Other physiochemical properties that have been employed in binding site prediction include residue side chain polarity and charge, amphiphilicity index, etc.

1.6.4. Evolutionary conservation. Evolutionary conservation of an amino acid in a family of proteins has been used as an indicator of functionally important sites within a protein. Interface residues tend to be more conserved than other surface residues. However, the difference in the degrees of conservation of interface residues and those in the interior of the protein is not large [34, 35]. Evolutionary features used in binding site prediction methods include sequence profiles, Position Specific Scoring Matrices (PSSMs), residue conservation scores, conservation of physiochemical properties, etc. A good review of different methods to score residue conservation is given in [36].

1.7. Anatomy of a protein binding site

Residues in protein interfaces can be divided into two major, at times overlapping, types: specificity determining residues and affinity determining residues. *In vivo*, proteins can be surrounded by many potential binding partners. However, a small set of specific non-covalent interactions encoded in the structure results in specific binding only to evolutionarily determined unique partners. The residues responsible for producing these interactions are called *specificity determining residues*. The prevailing theory explaining protein binding specificity introduces the concept of *anchor residues* [37] (see Figure 1.3). These residues, usually polar or charged, tether into a binding groove on the other protein in the initial stages of complex formation. This provides a geometric constraint that helps create a weak initial *encounter complex*. It has been observed that side chain conformations of anchor residues do not change much upon binding. These residues also bury the largest ASA after complex formation. Once these anchors are docked, flexible, solvent-exposed side chains latch to the encounter complex in the periphery of the binding pocket to form the final high-affinity complex. These latch residues are usually hydrophobic (e.g., Tryptophan, Phenylalanine,

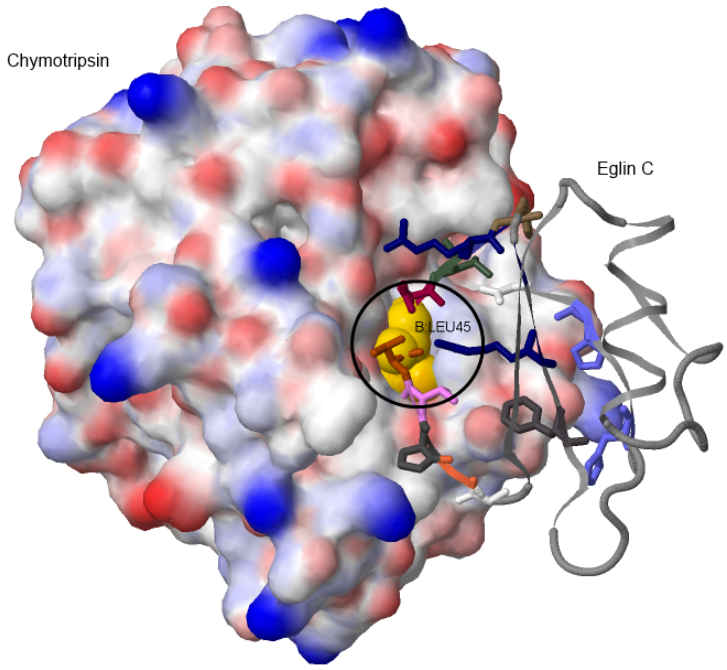


FIGURE 1.3. Anchor residue, Eglin C : LEU 45, in the α-Chymotrypsin/Eglin C complex (PDB: 1ACB [38]) shown in yellow space filled representation (encircled). 93% of the ASA of this residue is buried upon complex formation and its contribution to the binding free energy is -8.5kcal/mol. Other residues that get buried as a consequence of complex formation are shown as wireframe with the backbone of the rest of the protein in cartoon representation. The surface of Chymotrypsin is shown with colors indicating charge (red being positive and blue being negative). The figure was developed using the webserver ANCHOR [39].

etc.) and undergo a larger change in their side chain orientation on binding in comparison to anchor residues. Residues that confer binding strength or affinity to the complex are called affinity determining residues. Some residues, called *hot spots* [1], contribute significantly more to the binding affinity than other residues involved in the binding.

Unbound proteins are mostly surrounded by water molecules, and complex formation usually involves exclusion of water molecules from the interface. However, some water molecules remain in the interface and are thought to mediate hydrogen bonds between polar residues [1, 40]. Presence of water molecules in the interface extends the range of interactions between residues and also adds to the structural flexibility of the complex.

CHAPTER 2

Problem formulation and literature survey

The computational prediction of protein binding sites from sequence or unbound structure is a hard problem. This chapter describes the challenges inherent in the problem itself and those associated with its computational aspects. Details about existing methods, together with their limitations and shortcomings are also presented. First, an abstract mathematical formulation of the problem is provided.

Given two proteins A and B, expressed in terms of their structural or sequence constituents (such as residues, surface patches, domains or motifs), the objective of developing a binding site prediction method is to find a function $f(\cdot;\boldsymbol{\theta})$ that scores the binding propensity of these constructs. The focus of this work is to generate predictions at the residue level only. Here, $\boldsymbol{\theta}$ represents the parameters of the function. This problem comes in two different flavors (see Figure 2.1):

Partner-independent prediction: Here, the objective is to predict binding site(s) on a protein without any information about its binding partner(s), i.e., predicting whether a residue a in a protein A is involved in an interaction with *any* other protein. The scoring function for this problem can thus be written as $f(a) = f(\boldsymbol{\phi}(a);\boldsymbol{\theta})$. Here, $\boldsymbol{\phi}(a)$ represents the information being derived from residue a on protein A for use in the predictor. An example of a method that produces such predictions is meta-PPISP [41].

Partner-aware or partner-specific prediction: Unlike partner-independent predictors, partner-aware methods use information about the specific binding partner of a protein. The objective of such predictors is to find whether a residue a in protein

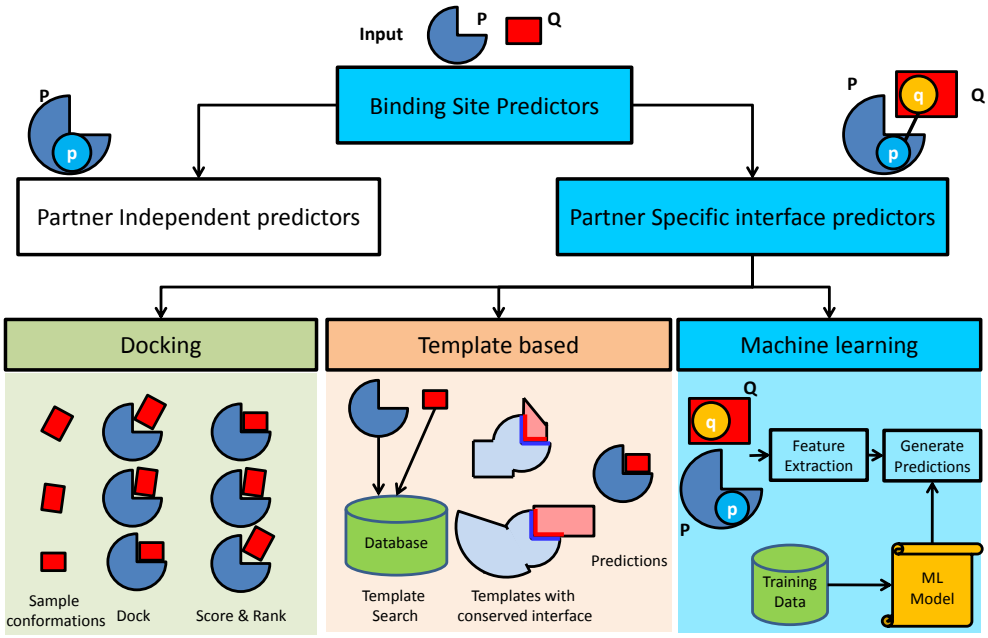


FIGURE 2.1. Different flavors of the binding site prediction problem; and different types of possible solutions. The output for each problem formulation is shown next to its box. The focus of this thesis is highlighted in cyan and steps in different schemes for partner specific interface prediction are shown.

A interacts with residue b in protein B upon the formation of the complex $A - B$. The scoring functions for such predictors generate pairwise binding propensities between residues in the two proteins forming the complex as: $f(a, b) = f(\boldsymbol{\phi}(a, b); \boldsymbol{\theta})$, where $\boldsymbol{\phi}(a, b)$ is the feature representation for the pair of residues from the two proteins. PPiPP produces such predictions at the residue level [42]. Moreover, docking methods such as ZDOCK can also be used to generate such predictions [43].

Thus, partner-independent predictors can only find protein binding sites whereas partner-specific predictors can provide information about both the interface in a complex and the binding sites on individual proteins forming the complex.

Depending upon what the objective is, what kinds of data is used, and what the exact formulation of the predictor is, the feature representation $\boldsymbol{\phi}$ in the above formulation can be modified accordingly. The parameters can be chosen empirically or estimated using

training data through machine learning techniques. The output of the scoring function can be class labels (interacting or not), numerical scores indicating the binding propensity or probabilities.

In the literature, partner-independent binding site prediction is the most well studied problem. However, the area of machine learning based partner-specific interface prediction is relatively unexplored and the accuracy of existing methods in this area is very low.

2.1. Challenges in predicting binding sites

The prediction of protein binding sites is made difficult by a number of factors. In this section, different challenges related to the problem and its solution are discussed [44, 45, 46].

2.1.1. Inherent challenges. Difficulties in the solution to the binding site prediction problem are introduced by the nature of the protein interactions as discussed below.

Dependence of binding propensity on the binding partner: The propensity of a residue in one protein to bind to another residue is dependent upon the nature of both residues (see Figure 2.2). This stems directly from the requirement of having physiochemical and geometrical complementarity between binding regions on the two proteins for the formation of a complex (see Section 1.3). Partner-independent methods ignore this fact while generating their predictions, whereas modeling this aspect of the problem in partner-specific predictors comes with its own challenges such as computational resource requirements, amount of available data, etc.

Alternative binding modes: The dependence of binding propensity on the binding partner allows proteins to use different binding regions to bind to different proteins [48, 49]. However, some proteins can bind to a number of other proteins

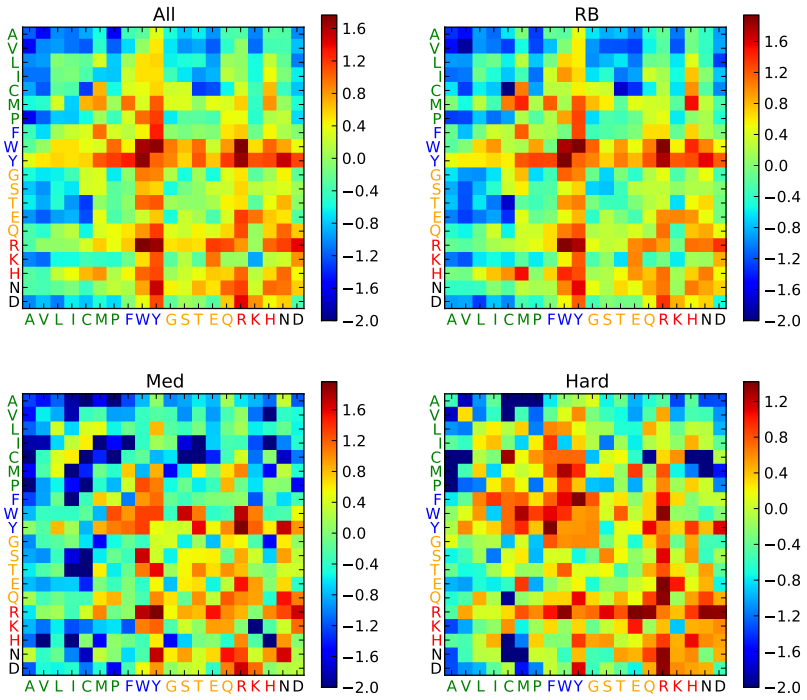


FIGURE 2.2. Binding propensities for different pairs of residues in complexes involving different degrees of conformational change in the interface (RB: Rigid body, Med: Medium difficulty and Hard). The colors of the amino acids on the labels of the x and y axes indicate their physiochemical properties (Non-polar: hydrophobic, aromatic and Polar: uncharged, positively charged, negatively charged). It can be noted that the preference of amino acids to bind to one another is dependent upon their physiochemical properties. It can also be seen that the binding propensity of a residue depends on the type of the residue in the other protein to which it binds. For example, R being positively charged has been observed to be involved in more interactions with the negatively charged D than with K which is also positively charged. The binding propensities of hydrophobic amino acids are lower because they are usually buried in the core of the protein whereas polar residues are exposed on the protein surface. It is interesting to note that the binding propensities also vary across complexes with different degrees of conformational change. For example, the propensities of hydrophobic-hydrophobic interactions are relatively higher in the Hard category in comparison to rigid body complexes. These propensities are calculated as the log_2 ratio of the frequency of pairs of binding residues to their expected frequencies based on the docking benchmark dataset (DBD 4.0) [47].

using the same binding region. A different but related issue is the possibility for structurally similar proteins to possess generally non-overlapping and significantly different interfaces even while binding similar partners [50]. However, enumerating all binding modes[4] of a protein or predicting its specific binding modes is challenging. In this context, partner-specific predictors can provide more informative predictions than partner-independent ones.

Protein flexibility and motion: Proteins are flexible entities that are in a constant state of motion. Protein flexibility is known to complicate the process of binding as it not only correlates with binding related conformational changes but also makes crystallization of the protein difficult for structural analysis [51]. Proteins that are very flexible need to be sufficiently stabilized (e.g., through mutations) before their structure can be determined using X-ray crystallography. Modeling the flexibility of proteins for binding site prediction is difficult because most of the available structure data is from X-ray crystallography which gives only a static snapshot of the protein. Protein flexibility can be studied either by using NMR structures or by predictions of molecular motion such as those obtained through Normal Mode Analysis [52] or molecular dynamics simulations, depending upon the time scale of the motion under investigation. One interesting and still open question is whether the flexibility of a residue has any direct correlation with its chances to be a part of the interface of a protein complex [53].

Conformational change: Proteins undergo conformational changes on complex formation. Such changes complicate the prediction of binding sites as complexes with

[4]A binding mode refers to a specific orientation of the one protein (ligand) relative to the other protein (receptor) in a complex as well as their conformations when bound to each other in a complex.

different degrees of conformational change employ different mechanisms for binding (see Figure 2.2). The prediction accuracy of most predictors decreases significantly with respect to the degree of conformational change [42]. Marsh and Tiechmann [51] have hypothesized that complexes with large conformational changes are at least 3 to 4 times more common than can be expected by looking only at the data for proteins which have both the bound and unbound structure available in the PDB. As a consequence, making accurate predictions for complexes involving large conformational changes is very important but also hard because of its inherent difficulty and lack of data.

Effects of long range interactions: A protein residue can interact with a residue far away from it in the protein sequence due to the folding of the protein chain into its three dimensional structure. Such long range interactions between residues far away from each other in sequence but in spatial proximity in the tertiary structure, make extraction of physiochemical properties difficult. This holds particularly true for sequence based features as it is easier to model and extract local properties (e.g., through sequence windows) than non-local ones [54]. As a consequence, these long range intra-molecular interactions complicate binding site prediction.

Diversity of protein complexes: The nature of the problem varies with the type of the protein complex. For example, homodimeric interfaces, on average, are more hydrophobic and much larger than heterodimeric ones [55]. Similarly, the binding patterns of obligate and transient complexes are significantly different. Among the different functional classes of protein complexes, antigen-antibody protein complexes are much more difficult to handle as antigens have highly variable regions [44, 42].

Developing a generalized predictor for all these diverse classes of complexes is difficult.

Discriminatory properties of binding sites: It would be easy to predict binding sites if binding residues had some clearly distinguishable property or set of properties in the unbound structure or sequence of the proteins involved in complex. However, no such set of properties is known to distinguish between binding and non-binding regions. Moreover, different properties of a residue are not independent of each other. For example, the flexibility of a residue is related to its depth within the protein which is also related to its degree of conservation as the core of a protein is usually more conserved than its surface. As a consequence, using multiple properties as features in a predictor may not essentially be adding significant information for the prediction.

Incomplete understanding of binding mechanisms: In the opinion of this author, the biggest problem in developing a binding site predictor is that not all aspects of the binding problem are thoroughly understood. For example, some proteins can interact with other proteins in multiple orientations [49, 56]. However, the exact mechanics of this process are not understood. Similarly, the role of water molecules in *wet* interfaces [40], post-translational modifications [57] and the effects of the environment [58] of a protein are only beginning to be understood and the amount of data available for such cases is very small.

2.1.2. Data challenges. The problem of binding site prediction is also made difficult by the amount, type and nature of the data available for solving the problem. Henceforth, different aspects of the problem with respect to the available data are presented.

Biases in data: The number of available X-ray protein structures in PDB is currently around 80,000 and has been increasing exponentially over the years, whereas the number of non-redundant structural folds[5] that are represented in it is only around 1,400 and has been stable for the last six years. Furthermore, the sampling of protein structures in the PDB is not a complete representation of the protein structural universe as protein structures cannot be determined for proteins that are flexible or intrinsically disordered. The structures of such proteins have been systematically missed [59, 44]. As a consequence, even a perfect structure based predictor of protein interactions trained or developed using existing PDB data will, at best, be able to solve only a subset of the problem. Moreover, the majority of proteins that undergo large conformational changes upon complex formation usually do not have their bound and unbound structures simultaneously available in the PDB [51]. Another bias comes from the fact that crystal structures usually record the most stable conformation of the protein even though the protein can exist in slightly different conformations in nature prior to binding. Such biases in the training data severely affect the generalization performance of binding site predictors, especially for cases which exhibit only a small amount of structural or sequence similarity to known protein complexes.

Noise in the data: Structural proximity thresholding is often used to define interacting protein residues. However, it is, at best, a proxy for binding as two residues within a small distance (say, 6.0 Å) of each other may not be truly interacting or their interaction may not be biochemically meaningful. For example, Gromiha et

[5]A *fold* is a certain way of arrangement of secondary structure elements in space which is repeated in a number of proteins.

al. [60] have found that, for their data set of 153 non-redundant hetero-dimeric complexes, 5.7% of residues within 6.0 Å of each other have strong repulsive energies and may not be truly interacting. This shows that there can be significant amount of noise in the labels used for training prediction methods, and that the interactions between residues should be defined with care, e.g., by considering the type of the two residues, their interaction energies, possible bond types and so on. Label noise can also stem from residue contacts due to crystal packing artifacts which are, for the most part, indistinguishable from true interactions [61, 55]. Noise can also exist in the three dimensional protein structures and features derived from it. Poor resolution structures contain significant amount of noise as well.

Amount of available data: Even though PDB has a large number of structures available, the amount of data usable in the development of an interaction prediction scheme is, by comparison, very small and it does not cover the universe of quaternary protein structures very well [62]. The data used in training binding site predictors needs to be non-redundant so as not to introduce any bias towards certain types of complexes, i.e., the set of protein complexes used in the training should cover the diversity of real protein interfaces. Garma et al. [62] have estimated that the total number of quaternary structure folds in the universe of protein complexes is around 4000 and the current PDB covers only 42% of them. Completing this coverage will require about a quarter of a century of experimental effort. Methods for interaction prediction require that both the unbound and bound structures of the proteins be available for all complexes in the non-redundant data set. This not only limits the number of available structures but also biases the data towards complexes with

small binding associated conformational changes. NMR structures can potentially provide more information. However, the number of good quality, non-redundant NMR structures of protein complexes in PDB is currently very small [47].

Selection of data sets and evaluation protocol: Different existing approaches create data sets with different parameters. For example, the redundancy thresholds, functional categorization of protein complexes in the data set, methods for defining interacting residues, etc., are very different for the training and validation data sets of different methods and this makes a fair comparison between existing methods or identifying a set of discriminatory features for classification very difficult. For example, some methods define patches on protein surfaces and the predictions are generated at the patch level [63] whereas others generate predictions for surface residues only [64]. Still others consider all (surface or non-surface) residues in the evaluation [42].

2.1.3. Modeling challenges. One of the most difficult task in computational prediction of binding sites is modeling of the problem itself. Although, all the information required for a protein to fold and to bind is contained in its sequence, extracting this information from sequence is not trivial. As a consequence, the accuracy of sequence-only predictors is lower than techniques that use structural information. Even with structure of the protein available, no known set of features, template database, energy function, ranking criterion or sampling technique is known to produce accurate predictions for a wide variety of protein complexes. Effective, efficient and accurate modeling of the problem is complicated by molecular motion, conformational change, long-range intra-protein interactions, shape irregularities and so on. Moreover, most existing methods make very limiting assumptions about

protein-protein interactions in order to simplify modeling. For example, some structure based methods model binding sites as circular patches [24] or sequence windows. However, this is incorrect because binding sites are mostly irregularly shaped and sequence windows cannot capture long-range interactions. Similarly, most existing methods do not consider the possibility of the presence of multiple partner-specific interfaces for a protein as they make partner independent predictions.

Another related issue is that of scalability. More accurate modeling of the problem usually increases the computational burden of the problem which limits the scalability and applicability of the method.

2.2. Existing methods

In this section we present a brief literature survey of existing methods for binding site prediction.

2.2.1. Partner independent predictors. A number of partner-independent methods have been proposed ranging from machine learning techniques to template or homology based approaches. A detailed review of these methods is beyond the scope of this dissertation as the primary focus of this research is partner-specific prediction. The interested reader is referred to [65, 44, 20, 66] for excellent reviews on the subject. Here, only a brief overview of these methods is provided.

Some of the currently best performing structure based partner-independent techniques that generate predictions at the residue level include meta-PPISP [41], PredUS [67] and PrISE$_C$ [68]. meta-PPISP combines the predictions from three predictors: cons-PPISP [69], Promate [27] and PINUP [32]. cons-PPISP [69] uses an ensemble neural network classifier with sequence profile and surface accessibility features. Promate [27] uses an empirical

scoring scheme based on physio-chemical properties, conservation, B-factors and geometrical features to predict binding sites. PINUP [32] uses an empirical energy function consisting of a side-chain energy term, a term proportional to solvent accessible area, and a term accounting for sequence conservation. These three methods have some complementary information in their predictions which is exploited by meta-PPISP to result in better prediction accuracy than the individual methods [41]. PredUS [67] is based on the idea that binding sites are conserved among a set of structurally similar proteins. The degree of structural similarity of a residue with a set of proteins with known binding sites is computed using local structural alignment. These similarity scores are then used to calculate a *contact frequency map* which, for each residue in the query protein, indicates the number of interface residues from other proteins that align to the residue of interest. A support vector machine is then trained to generate predictions with the contact frequency map and surface accessibility as features.

PrISE$_C$ [68] combines local structural similarity and similarity between protein surfaces of a query protein to similar structural elements from other proteins with known binding sites to generate prediction scores for each residue in the query protein. Jordan et al. [68] compare PrISE$_C$ with meta-PPISP and PredUS. On a data set of 56 unbound protein structures, the areas under the receiver operating characteristic curves (AUCs) at the protein level for these three methods are roughly comparable, with PrISE$_C$ performing marginally better than the others, with AUC ≈ 0.75.

2.2.2. Partner specific interface predictors. It is clearly evident from the discussion in section 2.1 that partner-specific predictors can be expected to be more accurate than partner-independent ones as the former present a more complete model of the phenomenon of protein binding.

To summarize, partner-specific predictors offer the following advantages over partner-independent predictors:

- Only partner-specific interface predictors can tell us what residues in one protein interact with what residues on the other protein.
- Partner-specific predictors can be used to enumerate the distinct binding modes of a protein in its interactions with other proteins whereas partner-independent predictors cannot provide such information.
- If two binding partners of a protein use the same binding site on the protein, then these binding partners cannot bind to the target protein simultaneously. Such information can only be predicted using a partner-specific predictor [48].
- Partner-independent predictors ignore the fact the binding propensity of a residue in a protein is also dependent upon the nature and local neighborhood of the residue to which it binds whereas partner-specific interface predictors can leverage this information to generate better predictions.
- Partner-specific predictions can be more readily incorporated in docking schemes and can result in more accurate docking than partner-independent ones.

Based on these potential advantages of partner-specific predictors, this thesis focuses primarily on these methods. Here, we discuss different categories of methods for generating partner-specific predictions and also describe specific methods within each category (see Figure 2.1).

Docking: The objective of macromolecular docking is to predict the three dimensional structure of a macromolecular complex given the unbound structures of its constituent molecules. Docking solves a more general and more complex problem

than binding site prediction. However, once the predicted structure of a protein complex is available from docking, the interface can be easily recovered. Almost all protein-protein docking techniques incorporate two key elements (see Figure 2.1) [70]: a *sampling algorithm* and a *scoring function*.

The objective of the sampling algorithm is to explore the conformational spaces of the unbound molecules. Most docking methods treat the unbound structures as rigid and sample a large number of their orientations using a relatively simple energy function to keep the method computationally feasible. This yields a set of candidate structures, in which the two partners contact each other without major steric overlaps and possibly possess desirable properties such as some level of shape, electrostatic, and chemical complementarity. Docking methods achieve such complementarity through FFT methods (e.g., ZDOCK [71], F^3Dock [72] etc.), Monte Carlo processing (e.g., RosettaDock [73]) or data-driven energy minimization (e.g., HADDOCK [74]).

However, the large number (in thousands depending upon granularity of the sampling) of complex structures produced by the sampling techniques described above generally exhibit limited accuracy, and the sampling needs to be followed by a scoring step, aimed at identifying near-native conformations of the complex. An example of such a scoring scheme is ZRANK [71] which models the electrostatics and effects of van der Waals forces to re-rank the initial stage predictions, and results in significantly more accurate predictions than the sampling procedure alone.

Docking methods are hampered by a lack of complete understanding of the forces involved in complex formation and by the conformational changes associated with

binding [44]. As a consequence, docking methods do not fare well in cases with large conformational change. For example, in a data set of 124 rigid and 52 non-rigid complexes, ZDOCK was able to find near-native structures in its top 10 predictions for 33 rigid-body complexes but for only two non-rigid complexes [71]. Docking methods can benefit from binding site predictions, as the correct identification of the interface can limit the degrees of conformational freedom for sampling. Some methods employ partner-independent predictors to accomplish this [75]. However, partner-specific interface predictions can be expected to play a better role [76].

Template based methods: With the growth in the number of protein complexes in the Protein Data Bank (PDB), both template-based interface predictors and template-based docking schemes have attracted attention. In these methods, a protein complex is modeled using sequence or structural similarity to a known template protein complex. Template based methods can use either homology derived from sequence or structure or interfacial similarity.

In homology based techniques, the assumption is that complexes formed with similar proteins have similar interaction modes. A recent sequence based non-parametric heuristic approach is PIPE-Sites [77]. PIPE-Sites is able to predict binding sites in proteins in a partner-specific manner from sequence information alone. PIPE-Sites does not involve any model learning or training and generates its predictions for a pair of sequence windows from two proteins in a complex by essentially counting the number of matches a given pair has in a database of known protein-protein interactions. A homology based strategy that uses protein structures for determining the three dimensional structure of protein complexes is presented in [78]. In this

method, the full structural alignment of proteins in a protein complex to a set of known protein complexes is used to construct a structure of the query complex. However, interface based methods outperform full structural alignment techniques. Interface based methods are based on the assumption that complexes with different proteins may have similar interface architectures. Thus, similarity to just the interface of known complexes, without global sequence or structural similarity, can be used for finding complex interfaces. Interface based methods generally exhibit better performance than homology based techniques. Examples of such methods include PRISM [79] and ISEARCH [80].

There are two important questions to ask in the context of template based methods: (I) How much sequence or structural similarity to template complexes is required for effective binding site prediction? and, (II) What extent of coverage of the known protein-protein interaction universe do template based methods currently provide? Aloy et al. have suggested that proteins that share more than 30-40% sequence identity generally exhibit similar interaction modes, especially if the proteins can be determined to belong to the same family. In terms of structural similarity, Kundrotas et al. [81] have found that complexes whose proteins have a minimum structural alignment score (measured using TM-align [82]) of more than 0.4, exhibit similar binding modes[6].

It has been estimated that sequence similarity based protein complex prediction can account for 15-20 % of known protein interactions [83]. For structure based techniques, Kundrotas et al. [81] have discovered that structurally similar template

[6]It is interesting to note that complexes with minimum protein sequence identity more than 40% typically have a minimum structural alignment score of more than 0.8. This clearly indicates that structural similarity has a more direct relationship to binding than sequence similarity.

complexes can be found in PDB for almost all complexes of structurally characterized proteins. For example, for a data set of 1296 new complexes deposited in PDB during 2009-2011, structurally similar templates were found for all but seventeen complexes in PDB released before 2009. However, only 28% of these were accurate enough to be used for further template based modeling.

A recent comparison between template based and docking methods shows that both types of methods have comparable performance and correct predictions of the two methods do not exhibit a large overlap indicating that these techniques are complimentary to each other [84]. One of the issues with template based techniques is that, at times, a small set of mutations can prevent binding from occurring and template based techniques generally fail in such cases.

Machine learning approaches: Direct prediction of interfaces using machine learning techniques is a relatively unexplored research area and the accuracy of existing methods in this category is low. Unlike template based methods, machine learning based techniques, depending upon the features used for the prediction, are also applicable in cases where template similarity of the query proteins to known interfaces cannot be established. In this work, we focus primarily on machine learning methods. One of the first such approaches is InSite [85]. InSite models interactions at the motif or domain level, and predicts pairs of interacting motifs that best explain a given protein-protein interaction network. As such, it does not use information about known interaction sites, and is limited by the richness of the motif library and its coverage across a given protein. Finally, it is more valuable to obtain binding

site information at the residue level, as it allows for a more detailed understanding of the interaction. Recently, Ahmad and Mizuguchi investigated the impact of performing partner-specific versus partner-independent binding site prediction [42]. Their analysis of the binding propensities of residue pairs in protein-protein interfaces clearly shows that the binding propensity of a residue is strongly dependent upon its partner in other proteins. On this basis, they hypothesized that considering residue pairs in interacting proteins for binding site prediction can improve performance, and found out that this is in fact the case. Their neural network ensemble predictor (PPiPP) uses position specific scoring matrix and amino acid composition features. However, the accuracy of PPiPP is low (AUC of 72.9). The sequence-based nature of their method allows the method to be applied to proteins for which only the sequence is known but at the same time it is unable to utilize the wealth of information contained in protein structures. To the best of the knowledge of this author, no structure-based partner-specific machine learning method exists for the solution of this problem.

2.3. Desired characteristics of a binding site predictor

In this section, we identify some of the characteristics expected from a 'good' interface prediction method. Please note that these characteristics are not independent of each other. These requirements will guide our efforts in model development, and can be useful for evaluating binding site prediction methods.

Accuracy: The notion of accuracy is important for any prediction scheme. However, in the context of binding site prediction, this notion is also dependent upon the intended use of the predictions being generated. Binding site predictors are frequently

used to identify probable binding site regions on proteins prior to experimental studies (e.g., mutagenesis experiments) in order to lower the amount of search involved in locating the true binding site. In this context, it is very important that the number of true positives in the top predictions should be high. For accuracy assessment and validation purposes, the degree of fulfillment of this requirement can be quantified by the rank of the first true positive prediction for complexes in the validation data set which should be low for a good predictor. The general behavior of a binding site predictor can be quantified using receiver operating characteristics (ROC) or precision-recall (PR) curves together with the area under these curves (AUC). However, these accuracy measures do not directly indicate the degree to which a predicted binding site can be meaningful in a biological or biochemical sense. For this purpose, it is often useful to analyze the spatial clustering of the top predictions along with other properties discussed below.

Spatial clustering: Residues forming the binding site in a protein complex are spatially clustered, i.e., they are usually closer to each other than randomly selected residues on the protein. For example, in the docking benchmark dataset (DBD 4.0) which consists of 175 non-redundant hetero-dimeric complexes [47], the average distance between binding residues (residues within 6.0 Å of the other protein) is 11.4 Å whereas the average distance between randomly selected residues is 20.1 Å. Thus, it is important that the top predictions generated by a binding site predictor be spatially clustered so as to define a biologically meaningful binding site.

Smoothness: In order to define meaningful binding sites, it is important that the predicted propensities should not change abruptly across nearby residues.

Sparsity: For any given protein, the number of binding residues is usually very small (typically, less than 10%). Thus the predictions should be sparse. This requirement is more important for interface prediction methods. This can be appreciated by the fact that in DBD 4.0, the average number of pairs of residues in two proteins forming a complex is 67,000 whereas the average number of interacting pairs is only 93.

Coverage, Applicability and usability: It is important that the predictor be able to handle a large variety of protein complexes with different biological functions, different degrees of conformational change and different degree of similarity to known protein interactions.

CHAPTER 3

PAIRpred: Partner-specific prediction of interacting residues from sequence and structure

In this chapter, we present our novel partner-specific protein-protein interaction site prediction method called *P*artner *A*ware *I*nteracting *R*esidue *Pred*ictor (PAIRpred)[7]. Unlike most existing machine learning binding site prediction methods, PAIRpred uses information from both proteins in a complex to predict pairs of interacting residues. PAIRpred captures sequence and structure information about residue pairs through pairwise kernels that are used for training a support vector machine classifier. As a result, PAIRpred presents a more detailed model of protein binding than partner-independent methods (see section 2.2.2 for more details). PAIRpred offers state of the art accuracy in predicting binding sites at the protein level as well as inter-protein residue contacts at the complex level. We demonstrate PAIRpred's performance on Docking Benchmark 4.0 and recent CAPRI[8] targets. We present a detailed performance analysis outlining the contribution of different sequence and structure features, together with a comparison to a variety of existing interface prediction techniques. We have also studied the impact of binding-associated conformational change on prediction accuracy and found PAIRpred to be more robust to such structural changes than existing schemes. As an illustration of potential applications of PAIRpred, we conduct a case study

[7]Our paper describing PAIRpred has been published in the journal *Proteins: Structure, Function and Bioinformatics* (2012 impact factor: 3.337) [86]. This work has also been presented at the Seventh International Workshop on Machine Learning in Systems Biology (MLSB) and Structural Bioinformatics and Computational Biophysics (3DSIG), Berlin, 2013 as well as 11th Annual Rocky Mountain Bioinformatics Conference, Aspen, Colorado, 2013. The code and evaluation data for PAIRpred is available online at: `http://combi.cs.colostate.edu/supplements/pairpred/`.

[8]CAPRI [87] is the acronym for Critical Assessment of PRediction of Interactions. It is a community wide experiment on the comparative evaluation of protein-protein docking for structure prediction

in which PAIRpred is used to analyze the nature and specificity of the interface in the interaction of the human ISG15 protein with the NS1 protein from influenza A virus.

3.1. Data and pre-processing

In the development of PAIRpred, we have used the protein-protein docking benchmark data set version 3.0 (DBD 3.0) [47]. This data set has also been used in the performance analysis of PPiPP [42] and allows a direct performance comparison. DBD 3.0 contains 124 non-redundant complexes of pairs of proteins for which both the bound and unbound X-ray crystallography structures are known. The proteins structures in DBD 3.0 have resolutions better than 3.25 Å and a minimum sequence length of 30. No two complexes in DBD 3.0 share the same SCOP [88] family-family pair [89] and have sequence identity of more than 30% in both chains. Further testing was performed on version 4.0 of DBD which contains a total of 176 complexes including those already in DBD 3.0.

3.2. Interacting residue-pair definition

We define two residues belonging to two different proteins in a complex to be interacting if the distance between any two heavy atoms of those residues in the *bound* conformations of their proteins is less than or equal to 6.0 Å. All other residue pairs from the two proteins on that complex were taken as negative examples. Similar definitions have been used in previous studies (see [42] and references therein). Defining interacting residues in this way resulted in a total of about 11,500 positive examples in DBD 3.0, i.e., 93 interacting residue pairs per complex on average. The average number of residues pairs, in overall, for a complex is around 67,000.

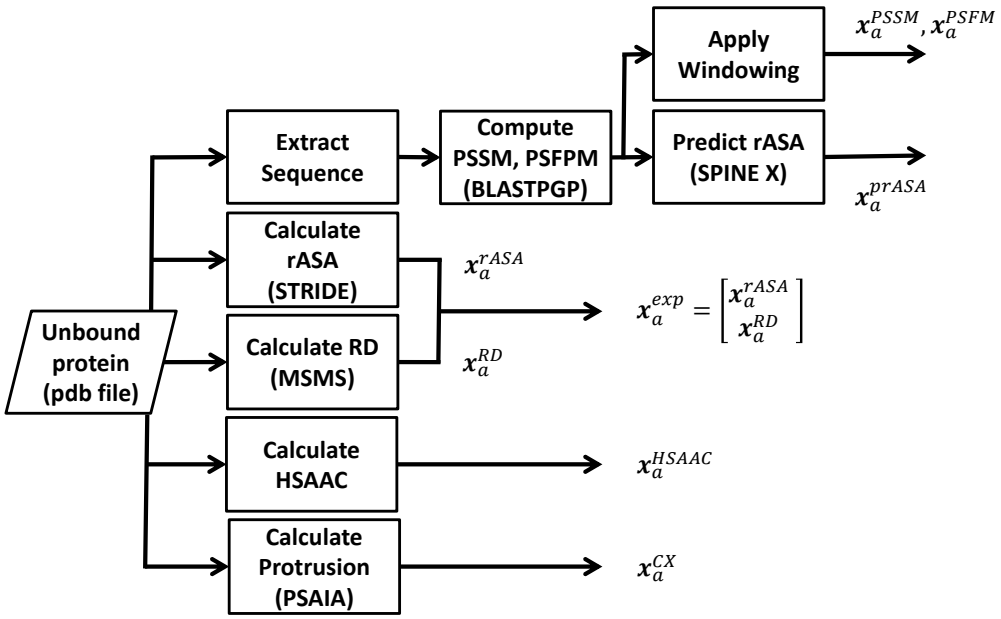


FIGURE 3.1. Residue-level feature extraction in PAIRpred. The feature representation for residue a is denoted by $\boldsymbol{x}_a$. Different components of the feature representation are denoted by the superscript (e.g., x_a^{rASA} indicates the relative accessible surface area for residue a). Each box also indicates the program used to extract a given set of features.

3.3. FEATURE EXTRACTION

We extracted both sequence and structure features at the residue level from the *unbound* structure of each protein. When the three dimensional of proteins forming a complex is not available, PAIRpred can make predictions based on its sequence alone. We have used a number of existing programs and methods from the literature to extract features from protein sequences and structures (see Figure 3.1).

3.3.1. STRUCTURE BASED FEATURES. The following features have been computed directly from the structure.

Relative Accessible Surface Area (x_a^{rASA})**:** The relative accessible surface area (rASA) from a given protein structure was computed using STRIDE [90].

Residue depth (x_a^{RD})**:** Residue depth is defined as the minimum distance of a residue from the surface of the protein and has been computed using MSMS [91]. The residue depth values produced by MSMS were normalized to have the range from 0 to 1. x_a^{RD} and x_a^{rASA} are combined to form a single surface exposure feature denoted by x_a^{exp}. We found that residue depth carries complimentary information to that in rASA for residue interaction prediction.

Half Sphere Amino Acid Composition (x_a^{HSAAC})**:** Hamelryck [92] found that the geometry and physiochemical characteristics of the regions in the direction of the side chain of a residue (called the 'up' direction) and in its opposite direction (called the 'down' direction) can be very different from each another. Based upon this observation, we computed a feature (called HSAAC) that captures the amino acid composition in the direction of the side chain of a residue $x_a^{HSAAC_u}$ and in the direction opposite to the side chain $x_a^{HSAAC_d}$. The amino acid composition in a direction is defined as the number of times a particular amino acid occurs in that direction within a minimum atomic distance threshold of 8.0 Å from the residue of interest. Thus, HSAAC combines surface accessibility and amino acid composition within the neighborhood of a residue. These amino acid composition vectors in the two directions are then normalized to have unit norm to get $x_a^{HSAAC_u}$ and $x_a^{HSAAC_d}$ which are then concatenated to get x_a^{HSAAC}. We utilized Biopython (Cock et al., 2009) to compute x_a^{HSAAC}.

Protrusion Index (x_a^{CX})**:** The protrusion index of a non-hydrogen atom is defined as the proportion of the volume of a sphere with a radius of 10.0 Å centered at that

atom that is not filled with atoms [93]. The protrusion index has been calculated using PSAIA [94].

The protrusion index for single residue is a 6 dimensional vector comprising the mean, standard deviation, maximum and minimum of the protrusion values of all atoms in the residue along with the mean and standard deviation of the protrusion values of only its side chain atoms. Each element of this vector is normalized to have the range from 0 to 1.

3.3.2. Sequence based features. We ran three iterations of PSI-BLAST [95] against the non-redundant 'nr' database [96] to compute the Position Specific Scoring Matrix (PSSM) and the Position Specific Frequency Matrix (PSFM) for a given protein. We also experimented with PSSM and PSFM extracted using HHblits [97] against the UniProt20 database which is a clustered version of the UniProt database with a maximum pairwise sequence identity of 20%. No major differences in accuracy, in comparison to the profiles from PSI-BLAST were observed. The following sequence based features are then computed from the profiles:

Profile Features (x_a^{PSSM}, x_a^{PSFM})**:** In order to extract the profile features for a residue from the PSSM, we took the PSSM columns within a length 11 window centered at that residue. This 20×11 matrix is converted to a single 220 dimensional unit vector denoted by x_a^{PSSM}. x_a^{PSFM} is constructed in a similar manner from the PSFM.

Predicted Relative Accessible Surface Area (x_a^{prASA})**:** To determine whether predicted rASA can be used instead of the true rASA, we used SPINE X [98] to predict rASA using the PSI-BLAST data. The predicted rASA is denoted by prASA to emphasize the fact that it has been predicted from sequence.

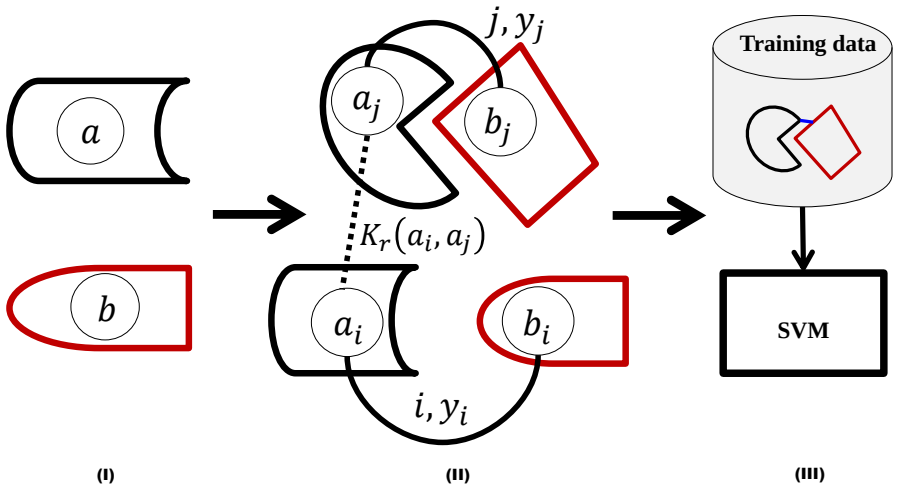


FIGURE 3.2. Overview of PAIRpred. (i) Extract residue-level features from sequence and unbound structures (see Figure 3.1 for details). (ii) Construct pairwise kernel from the residue-level kernel $K_r(a_i, a_j)$. (iii) Use the pairwise kernel to train the SVM and classify each residue pair in the query proteins.

3.4. PAIRWISE CLASSIFICATION USING SVMS

We model the interface prediction problem as a classification problem in which a classification example i is a pair of residues from two different proteins in a complex. Each example i is represented by $((a_i, b_i), y_i)$, where (a_i, b_i) is a pair of residues and y_i is the associated label, indicating whether the two residues interact ($y_i = +1$) or not ($y_i = -1$). Figure 3.2 illustrates this concept.

As a classifier, we use a support vector machine (SVM) [99] trained over a set of N labeled examples. An SVM finds the optimal separating boundary between two classes by simultaneously maximizing the margin between them and minimizing the cost of misclassification over the training data. For an overview of SVMs in computational biology, the interested reader is referred to [100]. Due to its large-margin nature, an SVM can offer good accuracy over previously unseen examples during testing.

In order to use an example in training or classification, a classifier needs the feature representation for the pair of residues in that example. However, it is easier and computationally more efficient to extract features for a single residue in a given protein than for a pair of residues from two different proteins. Thus, we would like to be able to use the residue level features directly to generate predictions at the residue pair level. This is where our use of SVMs for classification offers a significant advantage in comparison to other classifiers. Unlike other classifiers, such as the neural networks employed in PPiPP, SVMs can operate without requiring the explicit feature representation of an example by using a *kernel function* [100]. A kernel function is, in essence, a dot product that measures the degree of similarity between two examples. In this work, we employ *pairwise* kernels of the form $K((a,b),(a',b'))$ which can directly score the similarity between examples (a,b) and (a',b') by comparing the feature representations of individual residues in these examples. The pairwise kernel eliminates the need of constructing an explicit feature representation of each example because the scoring function of the SVM can be expressed only in terms of this pairwise kernel as: $f_{AB}((a,b)) = \sum_{i=1}^{N} \alpha_i y_i K((a_i,b_i),(a,b))$. In this scoring function, the values of α_i are obtained through training.

One of the interesting features of using pairwise kernels in the SVM is that these kernels can themselves be built from kernels over individual residues. Such residue level kernels, denoted by $K_r\,(a,b)$, compare the explicit feature representations of residues a and b to score the degree of similarity between them. The problem of constructing pairwise kernels from kernels over individual objects has been studied in the machine learning and Bioinformatics communities [101, 102, 103, 104]. We constructed the pairwise kernel K_{pw} for our SVM as

the additive combination of one or more of the following pairwise kernels from the literature:

$$K_{tppk}((a,b),(a',b')) = K_r(a,a')K_r(b,b') + K_r(a,b')K_r(b,a')$$

$$K_{mlpk}((a,b),(a',b')) = (K_r(a,a') - K_r(a,b') - K_r(b,a') + K_r(b,b'))^2$$

$$K_{sum}((a,b),(a',b')) = K_r(a,a') + K_r(b,b') + K_r(a,b') + K_r(b,a').$$

Here, K_{tppk} is the tensor product pairwise kernel (TPPK) proposed by Ben-Hur and Noble [101]. TPPK detects high similarity between examples (a,b) and (a',b') if a, expressed in terms of its feature representation, is similar to one of the residues in (a',b') and b is also similar to the other residue in the other example. It can be shown that the feature space of TPPK consists of products of features of the underlying residue kernel K_r.

$K_{mlpk}((a,b),(a',b'))$ is the metric learning pairwise kernel (MLPK) [102]. If the feature representation of a residue a is given by $\phi(a)$, then the MLPK kernel can be written as:

$$K_{mlpk}((a,b),(a',b')) = ((\phi(a) - \phi(b))^T(\phi(a') - \phi(b')))^2.$$

This shows that the MLPK is a homogeneous polynomial kernel of degree 2 between pairs after mapping a pair (a,b) to the vector $\Phi_{mlpk}((a,b)) = \phi(a) - \phi(b)$. Vert et al. have shown that MLPK performs slightly better than TPPK for predicting protein-protein interactions and that their additive combination performs better than either of the kernels [102].

Given the feature space representation $\phi(a)$ of a residue a, the direct sum pairwise kernel can be written as [103, 105]:

$$K_{sum}((a,b),(a',b')) = (\phi(a) + \phi(b))^T(\phi(a') + \phi(b')).$$

This shows that the sum kernel uses the underlying feature map $\Phi_{sum}((a,b)) = \phi(a) + \phi(b)$.

We found that the simple kernel K_{sum} performed better than both TPPK and MLPK for our problem. However, the additive combination of the three kernels performed better than any of the individual kernels (see the results section for more details). Finally, each pairwise kernel K_{pw} is normalized as $K((a,b),(a',b')) = \frac{K_{pw}((a,b),(a',b'))}{\sqrt{K_{pw}((a,b),(a,b))K_{pw}((a',b'),(a',b'))}}$ for use in the SVM.

To produce a pairwise prediction for an example (a,b), PPiPP [42] concatenates the feature representation of the two residues in the example in both orders $\begin{bmatrix}\phi(a)\\ \phi(b)\end{bmatrix}$ and $\begin{bmatrix}\phi(b)\\ \phi(a)\end{bmatrix}$. In comparison to PPiPP, our pairwise kernel based approach is computationally more efficient as it requires no duplication of the data. Moreover, pairwise kernels in our formulation directly model the inter-dependencies within individual feature components.

The residue kernel K_r used in constructing the pairwise kernel in PAIRpred is itself an unweighted summation of one or more of the following kernels, which are computed using the features described in section 3.3:

$$K_{profile}(a,b) = g(x_a^{PSSM}, x_b^{PSSM}; \gamma_{PSSM}) + g(x_a^{PSFM}, x_b^{PSFM}; \gamma_{PSFM})$$

$$K_{HSAAC}(a,b) = g(x_a^{HSAAC}, x_b^{HSAAC}; \gamma_{HSAAC})$$

$$K_{prASA}(a,b) = g(x_a^{prASA}, x_b^{prASA}; \gamma_{prASA})$$

$$K_{exp}(a,b) = g(x_a^{exp}, x_b^{exp}; \gamma_{exp})$$

$$K_{CX}(a,b) = g(x_a^{CX}, x_b^{CX}; \gamma_{CX}).$$

In the above equations, $g(a,b;\gamma) = exp(-\gamma\|a-b\|^2)$ is the Gaussian kernel. The parameter γ in the Gaussian kernel controls the decay of the exponential function. If γ is set too

high or too low, the exponential function can saturate at 0 or 1 which will inhibit effective learning from the training data. We chose the values of these parameters so that, for the majority of non-identical input vectors for a kernel, the similarity score does not saturate and maintains good dynamic range. This heuristic is inspired by the literature about parameter selection in radial basis function neural networks [106]. Once chosen in this manner, these parameters were not changed to optimize accuracy. The selected values of these parameters are as follows: $\gamma_{PSSM} = \gamma_{PSFM} = \gamma_{HSAAC} = 0.5, \gamma_{CX} = 1.0$ and $\gamma_{exp} = \gamma_{prASA} = 3.0$. As discussed in the results section, these values give good performance over test data. Training and classification has been performed using the SVM implementation in the machine learning library PyML [107].

3.5. Post-processing

A binding site or interface is a collection of spatially neighboring residues whose binding propensities are correlated (see section 2.3). Keeping this in mind, we smoothed the prediction score for a pair of residues by averaging prediction scores within their local neighborhoods through the following post-processing step:

$$f'_{AB}((a,b)) = \frac{1}{2}\left(\frac{\sum_{b' \in N(b)} f_{AB}((a,b'))}{|N(b)|} + \frac{\sum_{a' \in N(a)} f_{AB}((a',b))}{|N(a)|}\right), \quad (1)$$

where $f_{AB}((a,b))$ is the raw PAIRpred discriminant score from the trained SVM and $N(r)$ is the set of the 10 neighboring residues of residue r on the same protein including r itself. Thus the post-processed scores is the sum of the averages of the prediction scores of a residue on one protein with a set of residues on the other protein. As discussed in the results section, this simple post-processing scheme improves the prediction performance significantly.

3.6. Performance evaluation

Performance evaluation was carried out in two stages. In the first stage we compared different kernel designs, and residue-level features using five-fold cross-validation at the complex level. In this cross-validation procedure, examples from all complexes in our data set were divided into 5 folds such that all examples from a complex are found in exactly one fold. To reduce computational time during model selection, the 5 fold cross-validation was done using a class-size balanced sample from DBD 3.0 in which the number of randomly chosen negative examples for a complex is equal to the number of positive examples in it. For each fold, the value of the parameter C that controls the cost of misclassification over training data in the SVM was selected by performing a similar nested 5-fold cross-validation. The value of C was selected from $\{0.1, 1.0, 10.0, 100.0\}$. The classification function values and the known true labels of the examples were used to compute the Receiver Operating Characteristic (ROC) curve for each complex. The average of the area (expressed as a percentage) under the ROC curve for all complexes, labeled as AUC, has been used as the performance statistic for selecting the optimal model.

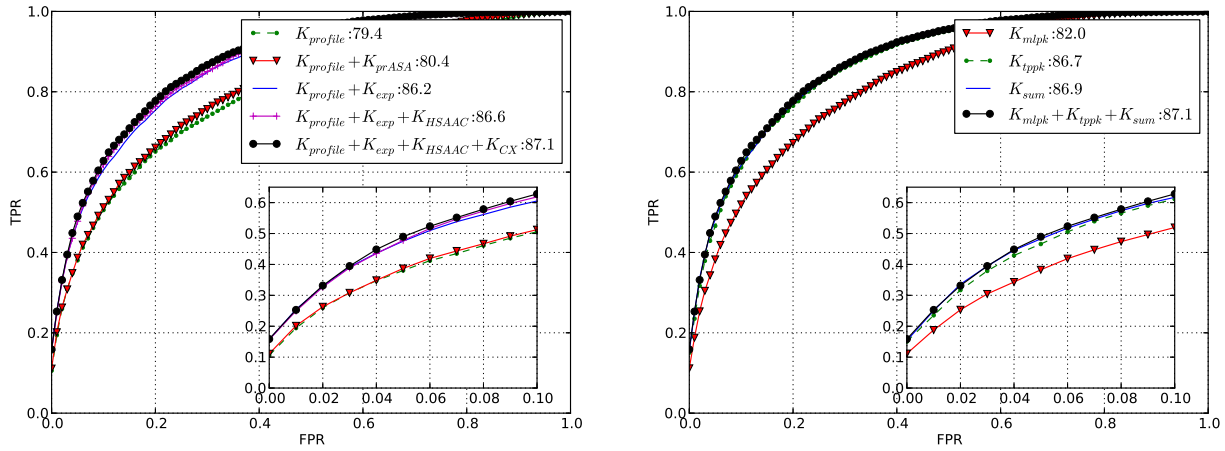
In the second stage of performance evaluation, we performed a leave-one-complex-out cross-validation analysis with the optimal kernel design selected in the first stage. In this cross-validation procedure, a classifier is trained on a balanced set of examples extracted from all but one of the complexes, and testing is performed on *all* pairs of residues from the left-out complex. This evaluation protocol is identical to the one used for PPiPP [42] and allows a direct and fair comparison between the two methods. The average area (expressed as percentage) under the ROC curves for all complexes (AUC) is used as a performance metric as it allows a quantitative comparison with other interface prediction methods. However,

AUC scores are not easy to interpret in this setting. In cases with highly unbalanced data with a big difference in the number of positive and negative test examples as we have here, AUC can give a false impression of accuracy. For these reasons we propose a measure of accuracy that is specifically designed for this domain. Our measure, which we call RFPP (rank of the first positive prediction), is defined as follows: RFPP(p) = q, if p % of the complexes tested have at least one true positive interacting residue pair among the top q predictions. Thus, an ideal classifier will have RFPP(100) = 1, i.e., in every complex, the top scoring prediction from the classifier belongs to the interface. In comparison to an ROC curve, this measure is more informative for the biologist as it tells us directly how often the top ranking predictions can be expected to correspond to known interactions.

We also evaluate the performance of PAIRpred for binding site prediction at the single protein level (i.e., binding site prediction) and compare it to existing partner-independent methods. Pairwise predictions of interacting residues at the complex level (from Equation (1)) are converted into predictions at the protein level for each protein as follows: $f_A(a_j) = \max_{b_j \in B} f'_{AB}((a_j, b_j))$ and $f_B(b_j) = \max_{a_j \in A} f'_{AB}((a_j, b_j))$. AUC scores for an individual protein can then be easily computed.

3.7. Results and Discussion

3.7.1. Comparison of residue and pairwise representations. We analyzed and compared different feature representations and pairwise kernel formulations in order to see the contribution of different features towards prediction accuracy and the impact of pairwise kernel design. Figure 3.3 shows the complex-wise averaged ROC curve for different feature and kernel combinations. In order to compare different feature representations, we chose to use $K_{pw} = K_{mlpk} + K_{tppk} + K_{sum}$ as the pairwise kernel. Our first step was to analyze the



(a) (b)

FIGURE 3.3. Selecting the optimal sequence/structure representation by comparing ROC curves for different kernel designs. Shown are the averaged ROC curves computed using 5-fold cross-validation over complexes in DBD 3.0. The inset shows the true positive rate (TPR) vs. false positive rate (FPR) for up to first 10 % false positives. The legend shows the AUC scores for the different kernels used. (a) Results for different residue kernels K_r using the pairwise kernel $K_{pw} = K_{mlpk} + K_{tppk} + K_{sum}$. The curves illustrate the increase in performance as additional structural information is added to the sequence-based kernel. Recall that $K_{profile}$ is the PSI-BLAST profile kernel; K_{prASA} uses predicted rASA; K_{exp} is the residue exposure kernel; K_{HSAAC} is the half-sphere exposure kernel; K_{CX} uses protrusion-index features. (b) Results for different pairwise kernels K_{pw} with residue kernel $K_r = K_{profile}+K_{exp}+K_{HSAAC}+K_{CX}$.

accuracy when our method is restricted to using sequence-based features only, which include sequence profile and relative accessible surface area predicted from sequence. As shown in Figure 3.3, profile features alone give an AUC of 79.4, and adding the predicted rASA (i.e., $K_r = K_{profile} + K_{prASA}$) increases the AUC to 80.4. For the profile-based features we found that the combination of PSSM and PSFM features performed slightly better than either of the two alone (results not shown). The addition of structure-based features provides a big boost in performance: the combination of true surface accessibility features (K_{exp}) with the profile features ($K_{profile}$) gives an AUC of 86.2 compared to 79.4 for the profile-based features alone and 80.4 using the combination of profile and predicted rASA features. Such

an improvement is to be expected because most of the residues involved in the interaction have high surface accessibility. However, the use of predicted rASA did not result in such a big increase. This is because the protein-wise averaged correlation between predicted and true rASA values for binding residues is low ($r = 0.56$, against $r = 0.76$ for non-interacting residues). Thus, the use of a better sequence based predictor of surface accessibility can help improve the accuracy of the sequence based predictions in future.

Addition of HSAAC and protrusion index based features ($K_{HSAAC} + K_{CX}$) improves the accuracy of the method even further (AUC of 87.1). For the rest of the analyses in the paper we have used $K_r = K_{profile} + K_{exp} + K_{HSAAC} + K_{CX}$.

The choice of a pairwise kernel has a strong influence on accuracy. Figure 3.3 shows the ROC curves for different pairwise kernel formulations. K_{mlpk}, K_{tppk}, and K_{sum} produce AUC scores of 82.0, 86.7, and 86.9 respectively, while adding all three provides an AUC score of 87.1. For the rest of the analyses in the paper we have used $K_{pw} = K_{mlpk} + K_{tppk} + K_{sum}$.

In order to test the variability of the results, the cross-validation procedure in model selection was repeated 5 times with change both in the randomly selected negative examples and the membership of complexes in different folds. We then evaluated the mean and standard deviation of different cross-validation runs. The maximum standard deviation in the AUC scores for any kernel combination was 0.2. This shows that these results are robust to changes in the training data.

3.7.2. Prediction using residue exposure alone. As discussed above, residue exposure features result in a big improvement in accuracy. To explore the contribution of the residue exposure features (rASA, reside depth, and mean protrusion), we computed the sum of the residue exposure of the two residues in each example. Using this combination

as a ranking criterion we computed the AUC score for each complex. This naïve way of classification yields some interesting results. The average AUC scores for all complexes from rASA, reside depth (RD) and the mean protrusion value are 71.9, 69.4 and 71.2 respectively. These results are only marginally inferior to the leave-one-complex-out cross-validation results from PPiPP (AUC = 72.9) [42]. Since rASA and RD are both measures of the surface accessibility of a residue, the AUC values for these features clearly reflect the known fact that surface residues are more likely to participate in protein-protein interactions. The AUC score of the protrusion index shows that the residues that interact have few atoms around them. This includes surface atoms, and especially those atoms on the surface that lie in cavities or protrude out from their local neighborhood. The protrusion index captures more local shape information than rASA and the two can be complementary to one another. The fact that pairwise summation of surface exposure features provides good results explains why the pairwise sum kernel K_{sum} was able to perform better than the other two pairwise kernels.

The same ranking criterion over the relative accessible surface area predicted from sequence using SPINE X gives an AUC of only 0.56. This clearly shows that these predictions need to be more accurate to be effective in finding interfaces in protein complexes.

3.7.3. Results for leave-one-complex-out cross-validation. For comparison with other methods we used the optimal kernel combination found through kernel evaluation (section 3.7.1) and recomputed its performance using the leave-one-complex-out cross-validation protocol detailed in Section 3.6. Results of this analysis are reported in Table 3.1. The AUC scores for interface prediction, averaged across the 123 complexes in DBD 3.0, for sequence and structure kernels are 80.9 and 87.3, respectively. It is interesting to note that

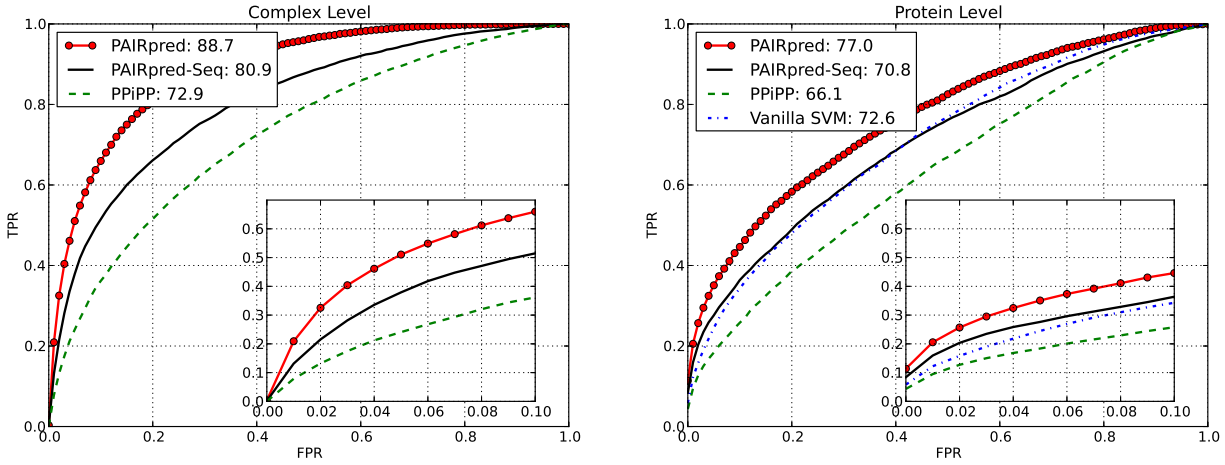
these scores from leave-one-complex-out cross-validation over all examples are very close to those obtained with the balanced sample. Evaluation over all the 176 complexes in DBD 4.0 gives an AUC score of 87.0 with the structure kernel. At the protein level, the AUC scores of PAIRpred for sequence and structure kernels for DBD 3.0 are 70.8 and 77.0 (with post-processing), respectively. It can also be noted that post-processing increases the performance of the method. This is particularly true at the protein level.

3.7.4. Comparison with PPiPP and ZDOCK. PPiPP [42] is a recently proposed sequence based method for partner-specific predictions that uses an ensemble of neural networks trained with a more elaborate version of our profile representation with different window sizes [42]. Table 3.1 shows the results of leave-one-complex-out cross-validation for DBD 3.0 using PPiPP. Even with the sequence features alone, PAIRpred gives better AUC and RFPP scores than PPiPP. As shown in figure 3.4, PAIRpred's performance at the complex level (i.e., for interface prediction) is superior to PPiPP not only in overall AUC but also in the number of true positives within the first 10% false positives.

PPiPP offers better accuracy than other published sequence based methods for binding site prediction such as PSIVER and SPPIDER (results given in [42]). PAIRpred's performance at the protein level (i.e., for binding site prediction) is also superior to PPiPP using either sequence features alone or in conjunction with protein structure (see Table 3.1 and Figure 3.4). We also compared PAIRpred with the docking method ZDOCK [71] over the 176 complexes in DBD 4.0. For this purpose, we have used, for each complex, the top 2000 predictions in the 15-degrees sampling data available online for ZDOCK v. 3.02. For each ZDOCK prediction for a complex, we computed the pairwise minimum inter-atomic distance between all residues of the two proteins in the predicted complex. The inverse of this distance

TABLE 3.1. PAIRpred and PPiPP performance. We compare the performance of PAIRpred and PPiPP [42] using Area Under the ROC Curve (AUC) and the rank of the first positive prediction (RFPP). RFPP(p) indicates that p percent of the proteins achieve that level of performance. For example, on DBD 4.0 without post processing, the second PAIRpred prediction is part of the interface for 10% of the complexes. PAIRpred results are provided for two residue kernels: the sequence-based kernel, and for the kernel that uses all the features computed from sequence and structure.

Dataset	Method		RFPP (p)					AUC	
			10%	25%	50%	75%	90%	Complex	Protein
DBD 3.0 (124 complexes)	PPiPP		9	19	78	297	760	72.9	66.1
	PAIRPred								
	$K_r = K_{profile} + K_{prASA}$	No post-processing	2	13	68	257	804	80.9	70.8
	$K_r = K_{profile} + K_{exp} + K_{HSAAC} + K_{CX}$	No post-processing	1	5	22	89	282	87.3	73.4
		With post-processing	1	3	16	103	272	88.7	77.0
DBD 4.0 (176 complexes)	$K_r = K_{profile} + K_{exp} + K_{HSAAC} + K_{CX}$	No post-processing	2	6	19	75	340	87.0	73.1
		With post-processing	1	3	18	101	282	87.8	75.4



(a) (b)

FIGURE 3.4. Comparison between PAIRpred, PPiPP [42] and vanilla SVM at the complex and protein level predictions on DBD 3.0. PAIRpred-seq refers to PAIRpred based only on sequence features.

was used as a ranking criterion in the evaluation of the AUC score at the complex level. The AUC score of a ZDOCK prediction tells us how good that prediction is at identifying the known interface in the complex and is directly comparable to the AUC scores given earlier for PAIRpred and PPiPP. For a given complex, we computed the maximum AUC score in the top N ZDOCK predictions and then averaged these scores across all complexes for a given

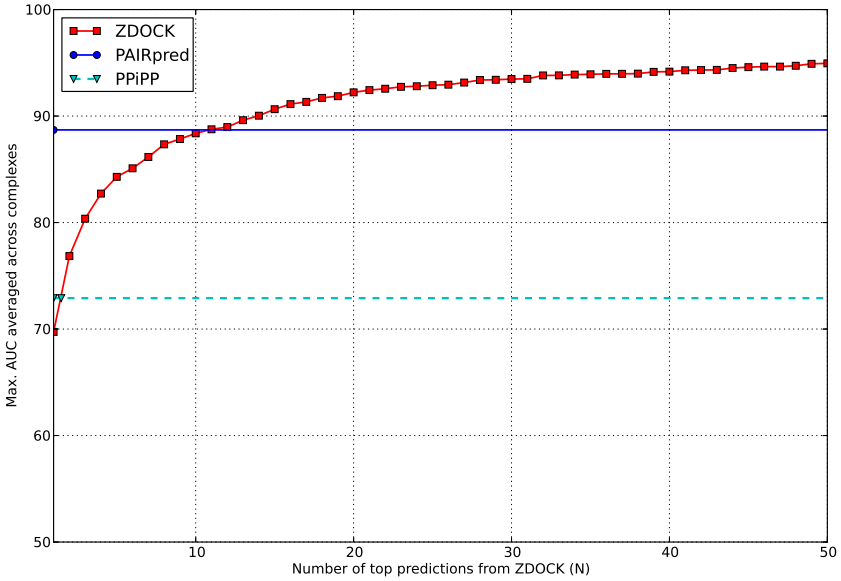


FIGURE 3.5. The maximum AUC from top N ZDOCK predictions in comparison to PAIRpred and PPiPP [42].

value of N to obtain the results shown in Figure 3.5. These results show that PAIRpred is better than the best of the top 11 ZDOCK predictions. The AUC score of the top prediction by ZDOCK is roughly comparable to that of PPiPP.

3.7.5. COMPARISON WITH PARTNER-INDEPENDENT PREDICTIONS. In order to test the hypothesis that a partner-specific predictor can perform better than partner-independent predictors, we developed an SVM based binding site predictor (referred to as *vanilla SVM*) using the same structural features as in PAIRpred and compared its leave-one-protein-out cross validation performance to the PAIRpred results at the protein level. Figure 3.4 shows the ROC curve for vanilla SVM which gives an AUC score of 72.6. PAIRpred performs much better than the vanilla SVM. This clearly shows that partner-specific predictors can offer superior performance in comparison to partner-independent ones even when the same residue level features are used. Moreover, PAIRpred's AUC score of 70.8 with the sequence

features alone is only marginally inferior to vanilla SVM even though the latter employs structure based features. As a matter of fact, PAIRpred with sequence features alone gives better true positive rates than the vanilla SVM consistently for false positive rates less than 0.4.

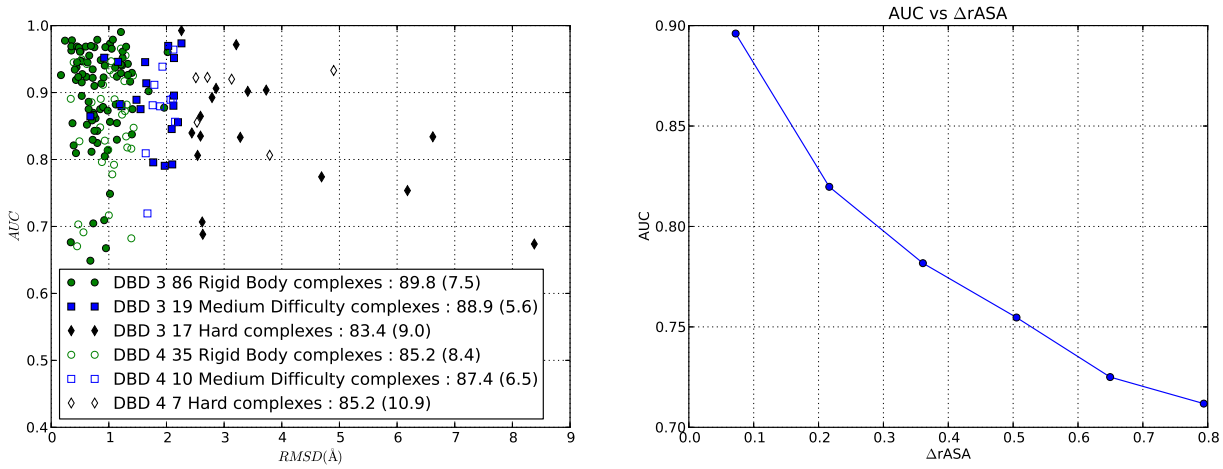
At the protein level, PAIRpred's performance using structure based features can be roughly contrasted to PredUS [67], a recently published structure based binding site predictor. PredUS performs better than other similar predictors available in the literature and gives an AUC score of 73.9 over 188 chains in DBD 3.0. It must be noted that a direct comparison between the performance of the two methods is not possible because of differences in their evaluation data sets, interface definitions, and cross-validation protocols. PAIRpred's performance with structure features can be expected to be equal or slightly better than that of PredUS as PAIRpred gives an AUC score of 77.0 over 248 proteins in DBD 3.0.

3.7.6. Spatial proximity of PAIRpred predictions. In order to see whether the top predictions by PAIRpred are spatially close, we compared pairwise distances between residues in our top predictions with a random sample of residues. More specifically, we computed the pairwise distances among the top 20 residue predictions from PAIRpred for each protein and also between the remaining pairs of residues from each protein. The average of the pairwise distances in the top predictions is 15.6 Å and 20.1 Å for the remaining pairs. These distances are significantly different (with a p-value of 4.7×10^{-25} using the Wilcoxon Rank Sum test on all complexes in DBD 4.0). This indicates the top PAIRpred predictions exhibit spatial clustering.

Furthermore, we found that the difference between the mean pairwise distances across the top predictions and the remaining residues in a protein is inversely correlated with the

its AUC (correlation coefficient of -0.49, 2 tailed p-value of 1.1×10^{-21}). Thus, this difference in distances is a rough indication of the quality of prediction.

3.7.7. Effects of conformational change. Proteins can undergo significant conformation change upon binding as buried residues can become exposed and vice versa. In order to observe the effects of the degree of conformational change on the accuracy of PAIRpred, we plotted the AUC of a complex against the root mean square deviation (RMSD) between the bound and the unbound states over the interface residue for in the complex. A large RMSD value for a complex corresponds to a large binding-associated conformation change. Figure 3.6 shows that the accuracy decreases with increase in conformational change. This effect was also observed for PPiPP. However, PAIRpred performs much better than PPiPP for complexes with large conformational change. Based on the degree of conformational change, the complexes in the docking benchmark datasets have been divided into three categories: rigid body, medium difficulty and hard. Figure 3.6 shows the prediction performance across complexes in these categories. As expected, PAIRpred performs better for rigid body complexes in comparison to the other two categories that involve larger conformational changes. We investigated the effects of conformational change on PAIRpred performance at the residue level as well. As we had access to both the bound and the unbound states of each protein, we were able to calculate the absolute difference in rASA for a residue between the two states of the protein. A large difference is indicative of a large conformational change in the environment around that residue. For a pair of residues we define the degree of conformational change as the sum of the changes in the individual residues, and denote it as $\Delta rASA(a, b)$. AUC exhibits a high negative correlation (see figure 3.6) with $\Delta rASA$ (correlation coefficient of -0.97, p-value of 1.5×10^{-3}). AUC vs. change in residue



(a) (b)

FIGURE 3.6. Effect of conformational change on PAIRpred performance. (a) AUC vs. RMSD for each complex in DBD 3.0 and the 53 new complexes in DBD 4.0 using leave-one-complex-out cross-validation. The legend shows mean AUC and standard deviation (within paranthesis) of complexes in each category for each data set. (b) Relationship between AUC and the change in rASA ($\Delta rASA$). Residue pairs were binned into groups based on their $\Delta rASA$ and the AUC score was computed for all the residues within each bin using our 5-fold cross-validation scheme on DBD 3.0 complexes.

depth shows a similar trend. This demonstrates the inherent difficulty of predicting residue-residue interactions in protein complexes that undergo a large conformational change. This difficulty is exacerbated by the fact that there is only a small amount of training data (24 complexes in DBD 4.0) available for such cases. Furthermore, the standard deviation of AUC scores for complexes from the hard category in DBD 4.0 shown in figure 3.6 is much larger in comparison to other categories. This suggests that effective handling of complexes with large conformational change requires a larger number of training examples with this property.

3.7.8. EVALUATION ON CAPRI TARGETS. In order to further analyze the performance of PAIRpred, we tested it on nine recent targets from the Critical Assessment of Protein Interactions (CAPRI) experiment [87]. We used all heteromeric protein complexes published

TABLE 3.2. PAIRpred evaluation results on recent heteromeric protein complex targets from CAPRI with both bound and unbound X-ray structures available. The degree of conformational change for the ligand and receptor proteins has been measured as the backbone root mean square deviation. The maximum sequence identity of a protein from the CAPRI set to any protein in DBD 4.0 has been calculated using local sequence alignment. The AUC score for each complex and the rank of the first positive prediction (RFPP) is reported.

Complex ID in PDB	Target ID in CAPRI	Ligand Backbone RMSD (Å)	Receptor Backbone RMSD (Å)	Max. Seq Id. of ligand to DBD4	Max. Seq Id. of receptor to DBD4	AUC	RFPP
4G9S	T58	0.3	0.7	28 %	27%	89.7	4
4EEF	T56	0.7	0.5	27 %	29 %	76.3	1
3R2X	T50	0.5	0.6	29 %	26 %	90.3	15
3U43	T47	0.9	1.5	60 %	55 %	88.9	2
2WPT	T41	2.0	0.7	62 %	66 %	85.8	1
3E8L	T40	0.2	0.4	100 %	28 %	92.1	9
3FM8	T39	0.0	1.6	28 %	25 %	79.6	71
3BX1	T32	2.0	0.4	30 %	56 %	89.7	10
2VDU	T29	1.1	0.4	28 %	27 %	82.9	302

after 2007 for which both the bound and unbound X-ray crystallography structures are available. For this task, PAIRpred was trained using DBD 4.0, and results of this analysis are reported in Table 3.2.

This table shows that PAIRpred is able to predict the interface with good accuracy for most targets. For seven out of these nine targets, the top 15 PAIRpred predictions contain at least one true positive. It is interesting to note that even for complexes involving large conformational changes, such as 3BX1 and 2WPT, the first true positive lies within the top 10 predictions. PAIRpred does not perform well on two targets: 3FM8 and 2VDU. These targets have proven to be very challenging for docking methods as well: only 1% and 4% of the models predicted by docking methods in CAPRI have an acceptable complex structure for 3FM8 and 2VDU, respectively [108].

3.8. Application to Human ISG15-Influenza A NS1 interaction

Due to its partner-specific nature and state of the art accuracy, PAIRpred can be used to study the nature and mechanics of an interface beyond what is possible with partner-independent predictors. In this section, we demonstrate PAIRpred's capabilities beyond the simple prediction of an interface by using the interaction between ISG15 protein in human and mouse and NS1 protein from Influenza A virus as a case study.

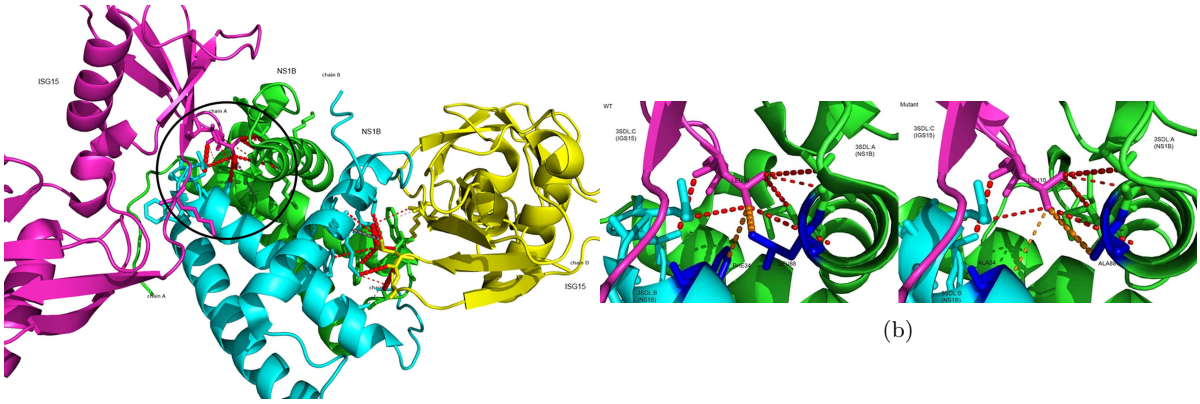
The influenza B virus is known to infect only human and non-human primates and the cause of this specific behavior have been investigated in [109] through a study of the bound and unbound structures of NS1 protein from the virus and the ISG15 protein in humans and other species. We have used PAIRpred to study the binding between these two proteins and compare the findings from this computational analysis to the results published in [109].

We first predicted the interface of the complex from the unbound structures of the two proteins using both PPiPP and PAIRpred and used the known interface to compare the performance of the two methods. The unbound PDB structures of NS1 and ISG15 are available as 1XEQ [110] and 1Z2M [111]. The complex structure (PDB ID: 3SDL) has two chains each of NS1 and ISG15 [112]. There is no significant conformational change in NS1 upon binding to ISG15 with only a disorder to order change in a short C-terminal polypeptide sequence. ISG15 undergoes modest conformational change upon binding NS1 with a backbone RMSD of 1.05 Å. We obtained the predictions from the unbound proteins by training PAIRpred on DBD 3.0 to allow for a comparison with PPiPP, and used structure-based features. This complex is not a part of training sets of PAIRpred or PPiPP. The AUC scores for PPiPP and PAIRpred for this complex are 67.2 and 92.4, respectively. The first true positive detected by PAIRpred is the top-most prediction, whereas the first true

positive detected by PPiPP occurs at rank 174. PAIRpred is able to find more than half of the interacting residue pairs within its top 100 predictions (see Figure 3.7). The predictions correspond very closely to the interactions discussed in [109]. We also compared the interface prediction performance of PAIRpred to that of ZDOCK for this complex by using the inverse of the inter-residue distance from ZDOCK predictions as a ranking criterion as described in Section 3.7.4. It was found that the AUC score from PAIRpred is better than the best of the top 13 ZDOCK predictions for this complex.

Next we used, PAIRpred predictions in order to identify the residues that are crucial for binding. Specifically, we conducted an *in silico* mutagenesis experiment in which we changed the NS1: L88 residue involved in our top prediction (ISG15: L10, NS1: L88) to an alanine. We also recapitulated one mutagenesis experiments reported in (Guan et al., 2011) which involved changing NS1: F34 (which also interacts with ISG15: L10) to an alanine. The (ISG15: L10, NS1: F34) interaction is originally ranked 8th in PAIRPRed predictions for this complex. We obtained the predicted structure after the mutations using I-TASSER (Roy et al., 2010). In comparison to the wild-type predictions for (ISG15: L10, NS1: L88) and (ISG15: L10, NS1: F34), we observed a decrease of 25% and 53% in prediction scores for L88 and F34 mutations in NS1, respectively (see Figure 3.7). The prediction scores for other interacting residues were essentially unchanged. These results indicate that both these residues are, as experimentally determined in [109], very important for this interaction.

As stated earlier, NS1 binds specifically to ISG15 from human and non-human primates and does not bind to mouse ISG15. Guan et al. [109] attribute this binding specificity to residues 47-52 and 76-80 in the sequence alignment of ISG15s from these three species. We obtained the unbound structure of mouse ISG15 using I-TASSER. We then compared



(a) (b)

FIGURE 3.7. (a) PAIRpred predictions for human ISG15 and influenza B NS1 mapped onto the 3D structure of the complex (PDB ID: 3SDL). The red dotted lines indicate the true positives in the top predictions with the width of the line proportional to the prediction score. The circled area is expanded in (b). (b) Results of in silico mutagenesis. The blue residues were changed to alanines. Notice the change in the prediction score (indicated by the width of the orange dotted lines) for the mutated residues between the wild-type (left) and the mutant (right).

the PAIRpred prediction scores for (human ISG15,NS1) complex to those from the (mouse ISG15, NS1) interaction. This comparison allowed us to identify the ISG15 residues that are interacting in (human ISG15, NS1) complex but undergo a large decrease in their prediction scores in the (mouse ISG15, NS1) interaction. These locations (in order of decreasing magnitude of change in predictions scores) are 76, 77, 72, 74 and 49. This strengthens the claim made in [109].

These analyses clearly demonstrate the usefullness of partner-specific predictions generated from PAIRpred as the mutagenesis studies explained above cannot be performed with conventional partner-independent predictors.

3.9. Using PAIRpred

PAIRpred has been implemented in Python and its architecture allows future extensions to include additional residue-level features or pairwise kernels. Complete implementation of PAIRpred, together with the pre-trained classifier, can be downloaded at `http://combi.cs.colostate.edu/supplements/pairpred/`. PAIRpred users need to supply the sequences in FASTA format or, when available, the PDB format structure files as input. PAIRpred then automatically extracts features from these files and produces predictions using a pre-trained SVM. Users also have the option of training the classifier on their own data sets. PAIRpred generates its prediction for a complex as a text file which contains the pairwise interaction scores for each pair of residues from the two query proteins. This pairwise prediction file can then be used to generate protein-level binding site predictions through scripts available as part of the PAIRpred package. PAIRpred implementation also provides PyMOL scripts for visualizing top PAIRpred predictions both at the complex and protein levels as shown in Figure 3.7.

3.10. Conclusions

We have presented a new method for predicting the interface of a protein complex called PAIRpred that offers state-of-the-art accuracy for both interface and binding site prediction. The proposed scheme is able to make accurate predictions using either sequence information alone or in conjunction with structure-based features. There are very few machine learning based methods that perform partner-specific prediction of interactions, and PAIRpred provides a large improvement over the recently published PPiPP method. We investigated the merit of sequence and structure-based features and found that using structure provides a big improvement in performance. Furthermore, the analysis of the accuracy of PAIRpred shows

much better scaling of performance with respect to the degree of conformational change upon complex formation in comparison to PPiPP.

Although, PAIRpred offers state of the art prediction accuracy, there is still significant room for improvement. This is particularly true for protein complexes whose formation involves large structural conformation changes in its constituent proteins. In the next two chapters, we describe our experiments with using structural alignments based features and more sophisticated machine learning schemes in an effort to improve the quality of prediction even further.

CHAPTER 4

Structural alignment and template based features for interface prediction

Structural alignment is a valuable tool in identifying homology between proteins, especially in cases where such evolutionary relationships cannot be established through sequence alignment techniques. This is because protein structure is significantly more conserved than protein sequence. Structural similarities between proteins can be used for prediction of interfaces because there is significant evidence that binding sites and interfaces are more conserved than other surface residues. We have developed three different types of features based on structural alignments of proteins that can be used in PAIRpred. These features capture structural conservation, local geometric similarity and similarity to a set of template complexes. In this chapter, we discuss the construction and performance evaluation of PAIRpred with these features.

Given the structures of two proteins, the task of a structure alignment algorithm is to find similarities between the two proteins and produce their pairwise alignment that optimizes a performance metric such as the root mean square deviation (RMSD) for the aligned structures. Examples of such algorithms [113] include Multiprot [114], TM-align [82], ProBiS [115], etc. We have used ProBiS in this work because, unlike most other commonly used local structural alignment methods such as Multiprot and TM-align, ProBiS produces accurate alignments in cases where the protein structures exhibit significant flexibility [113]. An example of a local structure alignment, obtained using ProBiS [116], is shown in Figure 1.2. ProBiS also produces e-values for each alignment which indicates the probability of

the alignment to occur by chance. In this work, we consider an alignment to be significant only if its e-value is less than 0.001. For more details on ProBiS, the interested reader is referred to Appendix A and [115, 117]. On the basis of such structural alignments, we extract three different classes of features for inclusion in PAIRpred. These features are described in detail in the following sections.

4.1. Structural conservation features

Protein structure is more conserved than sequence (see Section 1.2.2). As a consequence, structural alignments can be more sensitive in finding evolutionary relationships between proteins in comparison to sequence alignments. Local structure alignment of a given protein against a non-redundant set of proteins can be used to measure the degree of structural conservation of different residues on the protein. This set should, ideally, be a uniform sample set of the universe of proteins so the features are not biased towards any particular type of proteins. If certain regions on the protein show significant structural conservation against the non-redundant set, then those regions can be expected to be functionally important and may participate in binding to other proteins, ligands (small molecules) and ions. Figure 4.1 shows the residue-level structural conservation scores of a protein obtained through its pairwise local structural alignment against a set of approximately 33,000 non-redundant proteins in PDB (called NR-PDB). It is interesting to note that interface residues have significantly higher conservation scores than most non-interface surface residues.

In this work, we have used ProBiS to obtain local structural alignments of a protein against the NR-PDB. Based on these structural alignments, ProBiS produces a structural conservation score for all conserved residues in the query protein. Analogous to the position specific scoring matrix (PSSM) from PSI-BLAST, we constructed the PSSM for a query

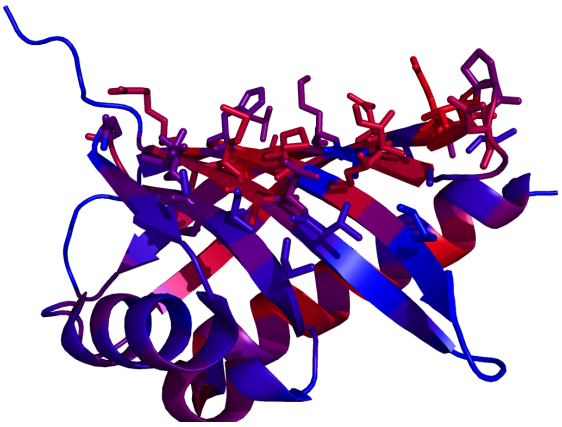


FIGURE 4.1. Interface residues in protein 1A2K (chain A) shown in stick representation with the color indicating the degree of structural conservation of the protein (blue to red) obtained using ProBiS through pairwise local structural alignments of 1A2K with about 33,000 non-redundant protein structures from PDB. Please note that the majority of interacting residues have high structural conservation scores.

protein based on its structural alignments against proteins in the NR-PDB. The PSSM score for residue at position i and amino acid a indicates the log-odds of the residue at i being structurally aligned to a in the NR-PDB relative to background frequency of occurrence of a in the NR-PDB. The PSSM features $\boldsymbol{x}_a^{PSSM_{probis}}$ and the ProBiS structural alignment scores $x_a^{sas_{probis}}$ can be directly incorporated in PAIRpred through the following residue level kernel (see Section 3.4 for details):

$$K_{NRPDB}(a,b) = g(\boldsymbol{x}_a^{PSSM_{probis}}, \boldsymbol{x}_b^{PSSM_{probis}}; \gamma_{PSSM_{probis}}) + g(x_a^{sas_{probis}}, x_b^{sas_{probis}}; \gamma_{sas_{probis}})$$

4.2. LOCAL GEOMETRIC SIMILARITY

We have developed a novel way of measuring local geometric similarity between residues belonging to two proteins using their structural alignments against the NR-PDB. We use the alignments of a protein A to obtain a local geometry similarity (LGS) descriptor $\boldsymbol{x}_a^{lgs}$ for each residue a in the protein. The length of the vector $\boldsymbol{x}_a^{lgs}$ is equal to the number of

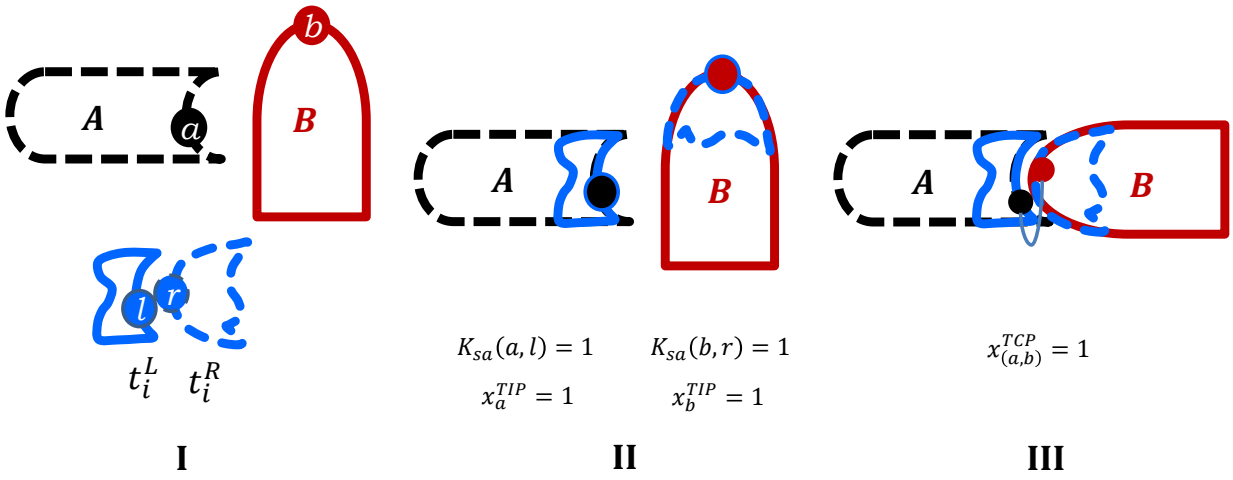


FIGURE 4.2. Computation of template interface and contact potentials based on the structural alignment kernel K_{sa}. Proteins in the query complex and a single template from the template set are shown in (I). (II) shows the structural alignment of A and B to the individual template components and the computation of the template interface potential using a single template. (III) shows the computation of the template contact potential.

proteins in the NR-PDB and $\boldsymbol{x}_a^{lgs}(P)$ is equal to 1 only if residue a is part of a statistically significant ProBiS alignment of A to P.

These descriptors can then be used to measure the geometric similarity between residues in two proteins because if two residues align to the same set of proteins in the NR-PDB, then it is highly likely that the two have significant local geometric similarity. The residue level kernel for local geometric similarity is a simple dot product kernel over the local geometric descriptors of the two residues: $K_{lgs}(a,b) = (\boldsymbol{x}_a^{lgs})^T \boldsymbol{x}_b^{lgs}$.

4.3. TEMPLATE BASED FEATURES

As discussed in Section 2.2.2, template based features can be quite helpful in the prediction of protein interfaces. In this section, we propose a scheme for incorporating such features in PAIRpred. Specifically, we propose two different representations of template based features: *template interface potential* (TIP) and *template contact potential* (TCP). In

this section, we describe both of these features in detail. The computation of these features assumes the availability of a non-redundant set T of template complexes.

In this work, we have used the template set from PRISM [118] with 1036 non-redundant templates. Each complex t_i in the template set is composed of two chains t_i^L and t_i^R. The set of pairs of interacting residues for t_i is denoted by $I(t_i) = \{(l,r)|l \in t_i^L \wedge r \in t_i^R \wedge d(l,r) < 6.0\ \mathring{A}\}$. Here, $d(l,r)$ is the minimum inter-atomic distance between residues l and r.

4.3.1. Template interface potential. Template Interface Potential (TIP) is a single residue level feature that measures the degree of similarity between a residue in a protein and all the interface residues in the template set. Mathematically, the general formula for TIP of a residue a can be written as:

$$\boldsymbol{x}_a^{TIP} = \sum_{t_i \in T} \sum_{(l,r) \in I(t_i)} K_r(a,l) + K_r(a,r). \tag{2}$$

In this formulation, $K_r(a,\cdot)$ is any similarity function that measures the similarity between a and a residue in a template interface. It can, possibly, be any of the sequence or structure based residue kernels discussed in section 3.4. However, we propose to use ProBiS alignments between a protein in a query complex against proteins in the set of templates to construct a similarity function K_{sa}. $K_{sa}(a,l) = 1$ only when ProBiS finds a significant alignment of a to l and zero otherwise. In essence, TIP represents a single residue level feature which can be easily incorporated into PAIRpred through a residue level kernel $K_{TIP}(a,b)$. This idea is illustrated in Figure 4.2.

4.3.2. Template contact potential. Template Contact Potential is a single residue pair level feature that measures the degree of similarity between a pair of residues from a

query complex and all interacting residues in complexes from the template set. Mathematically, the general formula for TCP of a pair of residues a and b can be written as:

$$\boldsymbol{x}_{(a,b)}^{TCP} = \sum_{t_i \in T} \sum_{(l,r) \in I(t_i)} K_r(a,l)K_r(b,r) + K_r(b,l)K_r(a,r). \tag{3}$$

The definition of K_r is identical to the one used in TIP. If we use the ProBiS based similarity function $K_{sa}(a,l)$ defined in the previous section in the above equation, then $x_{(a,b)}^{TCP}$ will be non-zero only if the residues a and b from the query complex simultaneously align to a pair of interface residues in at least one template in the template set. See Figure 4.2 for an illustration of this concept. It is important to note that TCP is defined at the level of pairs of residues and is, thus, a genuine pairwise complex-level feature which can produce a pairwise kernel directly: $K_{TCP}((a,b),(a',b')) = g\left(x_{(a,b)}^{TCP}, x_{(a',b')}^{TCP}; \gamma_{TCP}\right)$. This Gaussian kernel can then simply be added to the pairwise kernels (such as TPPK) computed using residue level kernels. This illustrates the flexibility of kernel methods as they allow easy and computationally efficient integration of different types of features.

4.4. Results

In this section, we analyze the effect of including the structural alignment and template based features described above on PAIRpred's performance. This evaluation has been carried out using leave one complex out cross-validation over the docking benchmark data set version 4.0 (DBD 4.0). However there are a few important differences between the evaluation protocol used here and the one described in section 3.7.3. Some proteins in DBD 4.0 contain multiple chains. A chain can potentially occur multiple times in a protein in DBD 4.0. and each 'copy' of the chain can use different interfaces in its interaction with another chain in

the formation of a complex. If we consider all unique pairwise interactions between chains in the 176 complexes, the resulting number of heterodimers in DBD 4.0 is 196. For this evaluation we performed cross-validation in a leave-one-complex-out fashion over these 196 hetero-dimers. The set of interacting residues for each dimer was developed by taking the symmetry and stoichiometry of the complex into account, e.g., if two 'copies' of a protein chain interact with another protein then the set of interacting residues is the union of the set of interacting residues of each protein chain. Since interacting residues in larger complex assemblies can be predicted by running PAIRpred over all interacting pairs of chains in the complex, the results presented here give a more realistic assessment of PAIRpred's performance in general. To establish a baseline, we first re-ran the best performing version of PAIRpred from Chapter 3 with this new protocol.

The results for different PAIRpred variants are given in Table 4.1. Note that with the changes in the evaluation data set describe above, the AUC scores go down slightly but the RFPP values show some improvement. The addition of structural conservation profiles and local geometric similarity features improve the RFPP scores further. The median RFPP with the addition of these features is 16 in comparison to 20 without them. The AUC scores remain roughly the same. This shows that these features do make the predictions more accurate.

The use of template based features warrants a more careful performance evaluation. This is because some of the proteins in DBD 4.0 exhibit significant sequence identity to the proteins in the template set. Calculating template based features for such proteins without considering their sequence identity to proteins in the template set will lead to overly optimistic prediction accuracies as in [64]. As a consequence, we computed template based

TABLE 4.1. PAIRpred evaluation results with structure alignment and template based features. In this table, $K_r = K_{profile} + K_{HSAAC} + K_{exp} + K_{CX}$ is used as the residue kernel to establish baseline results. This is the same residue kernel used for evaluation in Section 3.7.3. The evaluation has been performed using the new protocol described in the text with 196 hetero-dimers. $K_{NRPDB}, K_{LGS}, K_{TIP}$ and K_{TCP} are the kernels based on structural conservation (Section 4.1), local geometric similarity (Section 4.2), template interface potential (Section 4.3.1) and template contact potential (Section 4.3.2) features, respectively. SID is the sequence identity threshold in the evaluation of template based features. AUC is the area under the ROC curve calculated for both protein and complex level predictions. RFPP is the rank of the first positive prediction at different percentiles.

Kernel	RFPP					AUC	
	10	25	50	75	90	Complex	Protein
With original cross validation protocol over 176 complexes in DBD 4.0							
K_r	2	6	19	75	340	87.0	73.1
With new cross validation protocol over 196 heterodimers in DBD 4.0							
K_r	1	4	20	78	230	85.0	71.4
With structural conservation							
$K_r + K_{NRPDB}$	1	3	15	74	276	85.0	71.0
With local geometric similarity							
$K_r + K_{NRPDB} + K_{LGS}$	1	3	16	74	296	85.1	71.3
With template features $K_r + K_{NRPDB} + K_{LGS} + K_{TIP}$ and K_{TCP} added to pairwise kernel							
SID = 100 %	1	1	2	27	161	89.3	79.8
SID = 95 %	1	2	16	77	270	86.0	72.9
SID = 90 %	1	2	16	78	277	86.0	72.9
SID = 80 %	1	2	16	78	245	86.0	72.9
SID = 70 %	1	2	16	66	251	86.0	72.9

features with different sequence identity thresholds (SID). Specifically, during the computation of template based features for a particular protein or complex in DBD 4.0, only those templates are considered that do not have more than a certain sequence identity to the query protein. The sequence identities were obtained using local sequence alignments with the Smith-Waterman algorithm [119]. This allows us to understand the generalization characteristics of PAIRpred with template based features. As shown in Table 4.1, if no sequence identity thresholding is performed (SID=100%), we get excellent performance. However, this

improvement is because of complete sequence identity between the query proteins and the proteins in the template set. As SID is decreased, the number of proteins and complexes for which TIP and TCP are non-zero reduces significantly and the performance remains roughly the same. It is important to note that even at low sequence identities, the template based features seem to improve the AUC scores at both the protein and complex levels. However, the overall effect of including template based features and structural alignment features is not very pronounced. This may be because such features do not add more information for the prediction above and beyond the features already included in PAIRpred.

CHAPTER 5

Transductive and Semi-supervised Machine Learning Models for Interface Prediction

All existing machine learning methods for finding binding regions or interfaces in proteins, including PAIRpred, use supervised classification, i.e., they use a training set of known interacting and non-interacting examples from a number of complexes to induce a decision function to classify all query test cases. Each query example is treated independently of the other, even if they come from the same complex. As a consequence, such approaches ignore the fact that the classification examples from a query or test complex have inter-dependencies between them. For example, spatial clustering and sparsity requirements discussed in section 2.3 impose certain constraints on how examples in the query complex can be labeled.

In this chapter, we present machine learning techniques we have developed to model different types of inter-dependencies between examples from the query complex such as spatial clustering and sparsity. We incorporate these characteristics of the prediction problem by considering transductive [120] and semi-supervised classification schemes [121]. These schemes use unlabeled or partially labeled data from the test complex to develop a test-set specific classifier.

We have developed two separate classes of machine learning models for this purpose. Firstly, we experimented with 'wrapper' style algorithms that essentially use heuristics to manipulate the feature or kernel data and labels of the query examples to model dependencies between them while utilizing a classical support vector machine for classification. Our second approach models the inter-dependencies between examples from a query complex directly as

an optimization problem. The objective of optimization is to find a discriminant function which does not violate the inter-dependencies between examples of the query complex while maintaining good generalization characteristics and low empirical error on the training data. We formulate and propose a stochastic sub-gradient solver for this optimization problem. This chapter describes both the schemes in detail.

5.1. Transductive learning for interface prediction

PAIRpred, discussed in Chapter 3, is an inductive supervised classification scheme: it uses fully labeled training data to build a discriminant function which is then used for producing predictions for any example from any complex. The goal of training in such classifiers is to minimize classification error on the whole distribution of classification examples. However, using the same decision function for finding interfaces in a variety of query complexes can be one of the accuracy limiting factors in our problem domain because all types of complexes may not be represented or sampled equally well in the training data. This can be especially true for complexes involving large conformational changes for which the amount of available training data is small. A possible solution to this problem is to use *transductive learning* [120]. In transductive learning the learner receives as input both the training and test sets with the objective of learning a model to produce predictions for the test set with as few errors as possible *on that test set*. Transductive learning uses information from the unlabeled examples in the test set to improve the accuracy of prediction. Transductive learning and classification have been used in a variety of problem domains such as text classification [122, 123], image processing [124, 125], natural language processing [126, 127], etc. In Bioinformatics, transductive learning has been used to identify promoter regions [128], hot spot residues in protein complexes [129], protein classification based on phylogenetic

profiles [130], protein function prediction [131, 132] and literature mining [133]. To the knowledge of this author, no existing binding site or interface predictor is transductive in nature.

In this work, we have used the transductive support vector machines (TSVM) proposed by Joachims [134]. TSVM is a wrapper style algorithm and is well suited to our problem because it allows us to use pairwise and residue level kernels from PAIRpred directly. Moreover, this formulation will serve as a basis for modeling sparsity and geometric labeling constraints. In what follows, we describe the formulation of the interaction prediction problem as a TSVM.

TSVM requires a training set $S = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_N, y_N)\}$ of N labeled training examples from different complexes and an unlabeled test set $\{\boldsymbol{x}_1^\star, \boldsymbol{x}_2^\star, \ldots, \boldsymbol{x}_M^\star\}$ of M test examples from the query complex for which the labels $y_1^\star, y_2^\star, \ldots, y_M^\star$ are to be computed. We use $S^\star = \{(\boldsymbol{x}_1^\star, y_1^\star), (\boldsymbol{x}_2^\star, y_2^\star), \ldots, (\boldsymbol{x}_M^\star, y_M^\star)\}$ to indicate the set of test examples together with their labels which are to be obtained through transductive learning. The TSVM optimization problem can now be written as [134]:

$$\min_{\boldsymbol{w}, b, \boldsymbol{y}^\star, \boldsymbol{\xi} \geq 0, \boldsymbol{\xi}^\star \geq 0} \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C\sum_{i=1}^{N}\xi_i + C^\star\sum_{j=1}^{M}\xi_j^\star \tag{4}$$

Subject to:

$$y_i\left(\boldsymbol{w}^T\boldsymbol{x}_i + b\right) \geq 1 - \xi_i, \qquad \forall\,(\boldsymbol{x}_i, y_i) \in S \tag{5}$$

$$y_j^\star\left(\boldsymbol{w}^T\boldsymbol{x}_j^\star + b\right) \geq 1 - \xi_j^\star, \qquad \forall\left(\boldsymbol{x}_j^\star, y_j^\star\right) \in S^\star \tag{6}$$

$$\sum_{j=1}^{M} y_j^\star = 2M_+ - M., \tag{7}$$

where the parameters C and $C^\star$ control the impact of misclassification on examples from the training and test sets, respectively and the last constraint (Equation (7)) ensures that

M_+ test examples are classified as positive. This constraint can thus be used to control the number of positive examples in any complex. The solution to the above optimization problem produces not only the optimal weight vector $\boldsymbol{w}$ and bias b but also the labels $\boldsymbol{y}^\star = [y_1^\star, y_2^\star, \ldots, y_M^\star]^T$ for the test set while minimizing the error on both training and test sets.

The TSVM optimization problem described above is combinatorial in nature owing to the binary labels for the test examples, and as a consequence, convex optimization methods used in classical SVM cannot be employed here. For a small number of test examples, this problem can be solved by trying all possible assignments of the test set labels or using branch and bound style algorithms. However, this approach is impractical for the data sets in our problem domain. A number of algorithms have been developed to solve this problem. An excellent review and comparison of different algorithms has been presented by Chapelle et al. [135]. On the basis of the empirical results provided in [135], no single technique could be identified as consistently superior to others in terms of generalization performance. Therefore, we choose to use the algorithm originally presented by Joachims [134], and improved in terms of computational speed by Sindhwani and Keerthi [136]. This choice is based on the fact that this algorithm uses an out-of-the-box SVM formulation and offers comparable performance to more sophisticated algorithms in the comparison reported by Chapelle et al. [135]. Moreover, for the data sets considered in [135], this algorithm consistently performed better than a regular inductive SVM. This algorithm is based on a local combinatorial search guided by a label switching procedure which ensures that the value of the objective function does not increase and that convergence to a local minimum is reached in a finite number of steps. Algorithm box 1 gives a simplified pseudo-code of this algorithm.

Algorithm 1 Transductive SVM

Train a regular SVM over data in S to obtain the discriminant function.
Use the discriminant function to label examples in $S^\star$.
Set labels of top scoring M_+ test examples to $+1$ and of the remaining examples to -1.
Set $\overline{C} = 10^{-5}$
while $\overline{C} < C^\star$ **do**
  Solve the following quadratic optimization problem with fixed labels for the test set and $C^\star$ replaced by $\overline{C}$:

$$\min_{\boldsymbol{w},b,\boldsymbol{\xi}\geq 0,\boldsymbol{\xi}^\star\geq 0} \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C\sum_{i=1}^{N}\xi_i + \overline{C}\sum_{j=1}^{M}\xi_j^\star$$

  Subject to:

$$y_i\left(\boldsymbol{w}^T\boldsymbol{x}_i + b\right) \geq 1 - \xi_i, \qquad \forall\left(\boldsymbol{x}_i, y_i\right) \in S$$
$$y_j^\star\left(\boldsymbol{w}^T\boldsymbol{x}_j^\star + b\right) \geq 1 - \xi_j^\star, \qquad \forall\left(\boldsymbol{x}_j^\star, y_j^\star\right) \in S^\star.$$

  Obtain the discriminant function scores v_i and corresponding labels for all examples in the test set $S^\star$.
  Identify test examples with currently positive labels. Sort corresponding discriminant function scores in ascending order. Let the sorted list of examples be L_+.
  Identify test examples with currently negative labels. Sort corresponding discriminant function scores in descending order. Let the sorted list of examples be L_-.
  Set $U = \{\}$.
  Set $V = \{\}$
  for all i in L_+ **do**
    for all j in L_- **do**
      if $l\left(+1, v_i\right) + l\left(-1, v_j\right) > l\left(-1, v_i\right) + l\left(+1, v_j\right)$ and $j \notin U$ **then**
        $U = U \cup \{j\}$.
        $V = V \cup \{(i, j)\}$.
        Break.
      end if
    end for
  end for
  if V is not empty **then**
    Switch the labels of the pairs in V.
  end if
  $\overline{C} = 2 \times \overline{C}$.
end while
Return the current labels and discriminant function scores for the test set.

The algorithm, essentially, switches the labels of two examples from the query complex if doing so reduces the loss associated with margin violation. In this algorithm, we have used the hinge loss function $l\left(y, v\right) = \max\left\{0, 1 - yv\right\}$. This switching doesn't violate the

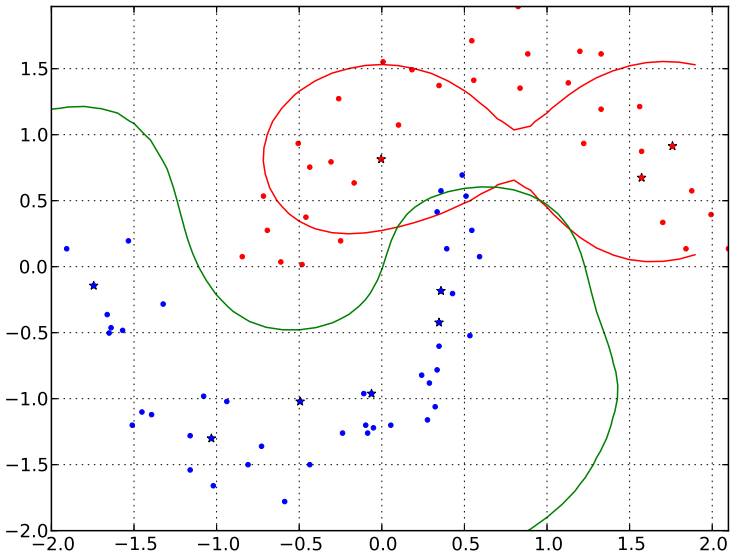


FIGURE 5.1. Illustration of the impact of transductive learning using the TSVM algorithm. Shown is a two-class toy data set from which only a few examples are used as labeled examples (marked with stars) to train a regular support vector machine and a TSVM. The decision function boundaries of the two classifiers are shown (red: regular SVM, green: transductive SVM).

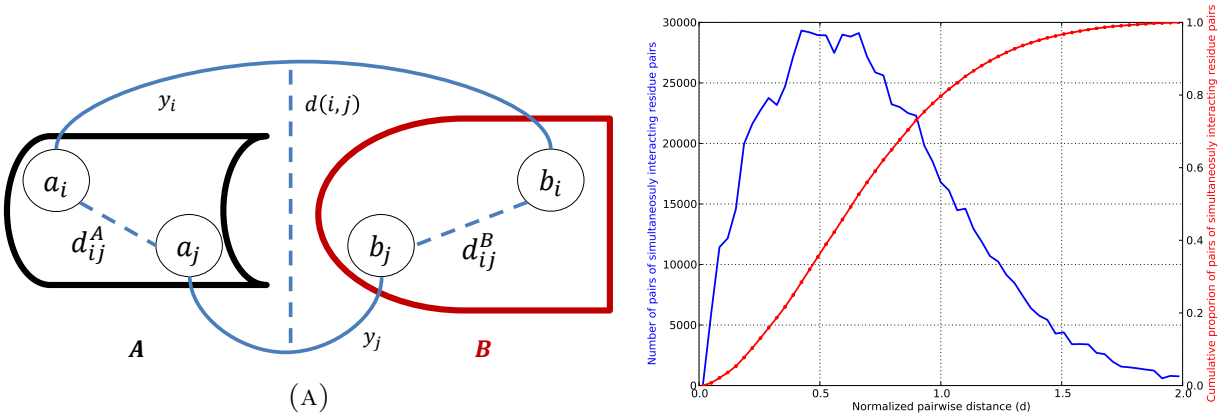
sparsity constraint at any time. This algorithm gradually increases the value of $\overline{C}$ in its iterations which controls the non-convex part of the objective function. This gradual increase, effectively, increases the impact of the test data on the decision function, and can protect the algorithm from converging to a sub-optimal local minimum. Figure 5.1 shows how this algorithm performs in practice on a toy data set. Note that the decision boundary in Figure 5.1 passes through a low-density region of the feature space. This behavior of the TSVM has been pointed out in the literature (see [135] and references therein).

5.2. Incorporating geometric labeling constraints

The three dimensional structures of proteins that form a complex impose constraints on which residues from two proteins can possibly interact with each other. For example, two

distant residues in a protein are not likely to simultaneously interact with another protein and even less likely to interact with the same residue on the other protein because this requires unrealistically large conformational changes on binding (see Figure 5.2a).

In this section we present an extension of the transductive SVM which explicitly models the geometric labeling constraint described earlier. This constraint can be written as: *whenever pairs of examples i and j in $S^\star$ are 'far away' from each other, at most one of the labels $y_i^\star$ and $y_j^\star$ can be $+1$* . Please note that a single example i is, in itself, composed of two residues in the two proteins forming the query complex. In order to formalize this constraint, we introduce a set $D^{AB} = \left\{(i,j) \mid d\,(i,j) > \theta_d^{AB}\right\}$ to store the indices of pairs of examples that are 'far away' from each other in a query complex composed of proteins A and B (see Figure 5.2b). In the figure, θ_d^{AB} is a tunable distance threshold and $d\,(i,j)$ is the normalized distance between the two examples i and j computed as $d(i,j) = \max(\frac{d_{ij}^A}{D^A}, \frac{d_{ij}^B}{D^B})$, where d_{ij}^A (or d_{ij}^B) is the distance between the residues in protein A (or B) that occur in examples i and j (see Figure 5.2a). D^A (D^B) is the median value of all pairwise inter-residue distances in protein A (B). These values allow us to normalize the distances within each protein with respect to the overall size of the protein. The geometric labeling constraint defined above, can now be mathematically expressed as: $\min\left(y_i^\star, y_j^\star\right) = -1, \forall\,(i,j) \in D^{AB}$. Figure 5.2b shows the histogram and cumulative distribution of normalized distances between interacting residues for all proteins in DBD 4.0. These plots show that the probability of two residues whose distance from each other is greater than twice the median distance to be involved simultaneously in an interaction with another protein is very small. This clearly shows that this condition can be used as a constraint to supplement transductive learning.



(A) (B)

FIGURE 5.2. (a) Defining labeling constraints for a query complex. (b) Histogram and cumulative distribution of normalized distances between interacting residues on a protein for all proteins in DBD 4.0.

The complete learning problem is expressed as:

$$\min_{\boldsymbol{w},b,\boldsymbol{y}^\star,\boldsymbol{\xi}\geq 0,\boldsymbol{\xi}^\star\geq 0} \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C\sum_{i=1}^{N}\xi_i + C^\star\sum_{j=1}^{M}\xi_j^\star \tag{8}$$

Subject to:

$$y_i\left(\boldsymbol{w}^T\boldsymbol{x}_i + b\right) \geq 1 - \xi_i, \qquad \forall\left(\boldsymbol{x}_i, y_i\right) \in S \tag{9}$$

$$y_j^\star\left(\boldsymbol{w}^T\boldsymbol{x}_j^\star + b\right) \geq 1 - \xi_j^\star, \qquad \forall\left(\boldsymbol{x}_j^\star, y_j^\star\right) \in S^\star \tag{10}$$

$$\min\left(y_i^\star, y_j^\star\right) = -1, \qquad \forall\left(i,j\right) \in D^{AB} \tag{11}$$

$$\sum_{j=1}^{M} y_j^\star = 2M_+ - M. \tag{12}$$

Addition of these distance constraints during learning entails some changes to the heuristic algorithm given in the previous section for TSVM. The pseudo-code of the modified algorithm is given in Algorithm box 2. Pairwise constraints have been utilized in other domains such as video object classification [137], clustering [138], etc.

Algorithm 2 Transductive SVM with geometric constraints

1: Train a regular SVM over data in S to obtain the discriminant function.
2: Use the discriminant function to label examples in $S^{\star}$.
3: Set labels of top scoring M_+ test examples to $+1$ and of the remaining examples to -1.
4: Set $\overline{C} = 10^{-5}$
5: **while** $\overline{C} < C^{\star}$ **do**
6: Solve the following quadratic optimization problem with fixed labels for the test set and $C^{\star}$ replaced by $\overline{C}$:

$$\min_{\boldsymbol{w},b,\boldsymbol{\xi}\geq 0,\boldsymbol{\xi}^{\star}\geq 0} \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C\sum_{i=1}^{N}\xi_i + \overline{C}\sum_{j=1}^{M}\xi_j^{\star}$$

Subject to:

$$y_i\left(\boldsymbol{w}^T\boldsymbol{x}_i + b\right) \geq 1 - \xi_i, \qquad \forall\left(\boldsymbol{x}_i, y_i\right) \in S$$
$$y_j^{\star}\left(\boldsymbol{w}^T\boldsymbol{x}_j^{\star} + b\right) \geq 1 - \xi_j^{\star}, \qquad \forall\left(\boldsymbol{x}_j^{\star}, y_j^{\star}\right) \in S^{\star}.$$

7: Obtain the discriminant function scores v_i and corresponding labels for all examples in the test set $i = 1, \ldots, \mid S^{\star} \mid$.
8: Identify test examples with currently positive labels. Sort corresponding discriminant function scores in ascending order. Let the sorted list of examples be L_+.
9: Set $\overline{S} = \{\}$
10: **for all** pairs of positive examples (i, j) from L_+ (with $i \neq j$) that violate the geometric constraint. **do**
11: **if** $v_i < v_j$ **then**
12: $y_i = -1$.
13: **else**
14: $y_j = -1$.
15: **end if**
16: Change the label of the currently highest scoring negatively labeled example to $+1$.
17: **end for**
18: $\overline{C} = 2 \times \overline{C}$.
19: **end while**
20: Return the current labels and discriminant function scores for the test set.

5.3. A stochastic sub-gradient optimization model for interface prediction

The transductive learning algorithm given in section 5.2 does not model the geometric labeling constraints directly in the objective function. The problem of semi-supervised classification with pairwise labeling constraints over unlabeled data has been a subject of interest for the machine learning community. A number of existing methods model this problem indirectly as a metric learning problem [139, 140, 141, 142, 143], while others propose a more

direct solution [137, 144, 145]. The pairwise constrained SVM presented by Nguyen and Caruana [145] is most relevant to our application domain. Their formulation assumes that a number of labeling constraints are available, which dictate whether two examples have the same label or not. In the context of our problem, the labeling constraints (see Equation (11)) are different from the ones in [145]. We propose a mathematical formulation inspired from the pairwise constrained SVM which directly minimizes an objective function including the pairwise constraints of Equation (11). Such an approach can be advantageous to the one presented in section 5.2 because direct learning ensures that the geometric labeling constraints from the test set are not violated. We also use the same formulation to enforce sparsity at the protein level, i.e., to ensure that the number of interactions for each residue on a protein is small.

In order to facilitate the development and explanation of this formulation, we choose to use a joint feature representation of classification examples which includes the feature vector of an example together with its label as $\boldsymbol{\phi}(\boldsymbol{x}^\star, y^\star) = \begin{bmatrix} \boldsymbol{x}^\star \mathbb{1}(y^\star = +1) & \boldsymbol{x}^\star \mathbb{1}(y^\star = -1) \end{bmatrix}^T$. In this representation, $\mathbb{1}(\cdot) = 1$ whenever the argument is true and 0 otherwise. The objective of this formulation is to learn the weight vector $\boldsymbol{w}$ which can be used in the discriminant function $f(\boldsymbol{x}) = \boldsymbol{w}^T \boldsymbol{\phi}(\boldsymbol{x}, +1) - \boldsymbol{w}^T \boldsymbol{\phi}(\boldsymbol{x}, -1)$ to generate the label for an example $\boldsymbol{x}$. Given a labeled training data set S and the test set $S^\star$ for which only labeling constraints D^{AB} are known (see section 5.2 for definitions), the learning problem considers the following factors:

Minimization of empirical loss: Like a classical SVM, our formulation also includes a term aimed at minimization of empirical loss over training examples from S. This is achieved by adding the term, $\sum_{i=1}^{N} l_S(\boldsymbol{w}; (\boldsymbol{x}_i, y_i))$ to the objective function. In this term, $l_S(\boldsymbol{w}; (\boldsymbol{x}_i, y_i))$ is the hinge loss function given by: $l_S(\boldsymbol{w}; (\boldsymbol{x}_i, y_i)) =$

$max\left\{0, 1 - z_i\right\}$ with $z_i = \boldsymbol{w}^T \boldsymbol{\phi}\left(\boldsymbol{x}_i, y_i\right) - \boldsymbol{w}^T \boldsymbol{\phi}\left(\boldsymbol{x}_i, -y_i\right)$. The loss function scores can be weighted differently for different examples in order to control effects of class imbalance. The minimization of this term leads to the satisfaction of the constraints: $\boldsymbol{w}^T \boldsymbol{\phi}\left(\boldsymbol{x}_i, y_i\right) \geq 1 + \boldsymbol{w}^T \boldsymbol{\phi}\left(\boldsymbol{x}_i, -y_i\right)$ for all training examples in S. This ensures that the examples in the training data are labeled correctly.

Margin maximization: A large margin between positive and negative examples ensures better generalization. As in a classical SVM, margin maximization is achieved in our formulation by minimizing the norm of the weight vector $\boldsymbol{w}$. Mathematically, the complete learning problem is represented as a minimization in which the objective function contains the term $\frac{\lambda}{2}\boldsymbol{w}^T\boldsymbol{w}$ for margin maximization. The parameter λ controls the extent of this regularization.

Satisfaction of geometric constraints: The geometric labeling constraints are modeled in our formulation by the addition of the term $\nu \sum_{(i,j)\in D^{AB}} l_{S^\star}\left(\boldsymbol{w}; \left(\boldsymbol{x}_i^\star, \boldsymbol{x}_j^\star\right)\right)$ in the objective function. Here, the loss function is given by: $l_{S^\star}\left(\boldsymbol{w}; \left(\boldsymbol{x}_i^\star, \boldsymbol{x}_j^\star\right)\right) = max\left\{0, 1 - z_{(i,j)}\right\}$ in which $z_{(i,j)} = \max_{\left(y_i^\star, y_j^\star\right)\in V}\left(\boldsymbol{w}^T \boldsymbol{\phi}\left(\boldsymbol{x}_i^\star, y_i^\star\right) + \boldsymbol{w}^T \boldsymbol{\phi}\left(\boldsymbol{x}_j^\star, y_j^\star\right)\right) - \left(\boldsymbol{w}^T \boldsymbol{\phi}\left(\boldsymbol{x}_i^\star, 1\right) + \boldsymbol{w}^T \boldsymbol{\phi}\left(\boldsymbol{x}_j^\star, 1\right)\right)$. The set $V = \left\{(+1, -1), (-1, +1), (-1, -1)\right\}$ contains all possible pairs of labels for examples in D^{AB}. The minimization of this term implies that for $(i, j) \in D^{AB}$, both i and j cannot be labeled as positive simultaneously and can only take a value from the set V. The parameter ν controls the relative importance of this term.

Sparsity at the protein level: To ensure that the number of interacting residues in each protein is kept small, two separate terms, one for each protein in the query complex, are added to the objective function. For protein A, this is achieved by addition

of the term $\frac{\mu}{|A|}\sum_{a\in A} l_A(\boldsymbol{w};a)$ to the objective function. In this term, the loss function is given by: $l_A(\boldsymbol{w};a) = max\{0, z_a - c\}$ with $z_a = \frac{1}{\mid E_A(a)\mid}\sum_{i\in E_A(a)} f(\boldsymbol{x}_i^\star)$, and $E_A(a)$ is the set of all test examples in $S^\star$ that contain the residue a from protein A. The minimization of this term leads to: $\frac{1}{\mid E_A(a)\mid}\sum_{i\in E_A(a)} f(\boldsymbol{x}_i^\star) \leq c$. This is a relaxed version of the exact sparsity constraint: $\frac{1}{\mid E_A(a)\mid}\sum_{i\in E_A(a)} \frac{y_i^\star+1}{2} \leq \overline{c}$ which allows each residue in A to interact with at most $\overline{c}$ residues in B. Analogous to $\overline{c}$, the parameter c in our formulation acts as an effective control for the proportion of interacting residues in A. The parameter μ controls the contribution of this term in the optimization.

Based on the discussion above, the complete learning problem can be expressed as the following unconstrained optimization problem:

$$\min_{\boldsymbol{w}} \frac{\lambda}{2}\boldsymbol{w}^T\boldsymbol{w} + \frac{1}{N}\sum_{i=1}^{N} l_S(\boldsymbol{w};(\boldsymbol{x}_i,y_i)) + \frac{\nu}{\mid D^{AB}\mid}\sum_{(i,j)\in D^{AB}} l_{S^\star}\left(\boldsymbol{w};\left(\boldsymbol{x}_i^\star,\boldsymbol{x}_j^\star\right)\right) + \frac{\mu}{\mid A\mid}\sum_{a\in A} l_A(\boldsymbol{w};a) + \frac{\mu}{\mid B\mid}\sum_{b\in B} l_B(\boldsymbol{w};b).$$

It is important to note that the last three terms in the formulation have the tendency to make the discriminant function scores of all examples from the test complex non-positive. To counter this, we also add a global sparsity constraint: $\frac{1}{\mid S^\star\mid}\sum_{i\in S^\star} f(\boldsymbol{x}_i^\star) = p^+$. This constraint is a relaxed form of the constraint in TSVM (Equation (7)). Note that if we make the mean of the feature representation of examples in the test complex equal to zero (i.e., $\frac{1}{\mid S^\star\mid}\sum_{i\in S^\star}\boldsymbol{\phi}(\boldsymbol{x}_i^\star) = \mathbf{0}$), then this constraint can be satisfied outside of the optimization procedure by including a bias term $\rho = p^+$ in the objective function as: $f(\boldsymbol{x}) = \boldsymbol{w}^T\boldsymbol{\phi}(\boldsymbol{x},+1) - \boldsymbol{w}^T\boldsymbol{\phi}(\boldsymbol{x},-1) + \rho$. A similar procedure for enforcing class balancing

constraints in TSVM learning is described in [146, 135]. This 'centering' can be implemented directly in kernels as well [147].

Now, we make some general observations on the formulation given above:

(1) If ν and μ are zero and $\lambda = \frac{1}{CN}$, the formulation reverts to a classical SVM with C as its cost of margin violation.

(2) The above function is strongly convex[9] with respect to $\boldsymbol{w}$.

(3) The problem is not continuously differentiable because of the presence of the hinge loss functions.

Due to the discontinuous nature of the objective function, we cannot use a gradient descent algorithm to solve this problem. Similar to the work by Nguyen et al. [145] and Shalev-Shwartz [148], we also propose a customized stochastic sub-gradient[10] optimization (SSO) based algorithm for the solution to this problem, which is presented in Appendix B.

Figure 5.3 shows the results of this algorithm on a two-class toy data set. In this example, the algorithm is provided with only three labeled examples (represented by squares) and a set of pairwise constraints of the form $\min\left(y_i^\star, y_j^\star\right) = -1$. The examples involved in forming the pairwise constraints are shown as triangles. The sparsity constraints are not used for this example (i.e., $\mu = 0$). Notice that the SSO-SVM with pairwise constraints is able to construct, near perfectly, the classification boundary (red). In comparison, the boundary

[9]A function $f(x)$ is called strongly convex if for all (x, y) in its domain, any $0 \leq t \leq 1$ and any $m > 0$: $f\left(tx + (1-t)\,x\right) \leq tf(x) + (1-t)f(y) - \frac{1}{2}mt(1-t)\|x-y\|_2^2$. Optimization of strongly convex functions using gradient descent and stochastic gradient descent is much faster than for convex functions that are not strongly convex or non-convex functions.

[10]A vector $\boldsymbol{\nabla} \in \mathbb{R}^n$ is called the sub-gradient of $f : \mathbb{R}^n \rightarrow \mathbb{R}$ at $\boldsymbol{x} \in dom\,(f)$ if $\forall \boldsymbol{z} \in dom\,(f)$, $f\,(\boldsymbol{z}) \geq f\,(\boldsymbol{x}) + \boldsymbol{\nabla}^T\,(\boldsymbol{z} - \boldsymbol{x})$, i.e., the affine function (of $\boldsymbol{z}$) $f\,(\boldsymbol{x}) + \boldsymbol{\nabla}^T\,(\boldsymbol{z} - \boldsymbol{x})$ is a global under-estimator of f. For example, the sub-gradient of $l\,(\boldsymbol{w}, \boldsymbol{x}) = max\left\{0, 1 - \boldsymbol{w}^T\boldsymbol{x}\right\}$ with respect to $\boldsymbol{w}$ at $\boldsymbol{w}$ is $-\boldsymbol{x}$ if $\boldsymbol{w}^T\boldsymbol{x} < 1$ and 0 otherwise. A function has a set of sub-gradients at a given $\boldsymbol{x}$, each element of which satisfies the above definition. $\boldsymbol{x}^*$ is a minimizer of a convex function f if and only if the sub-gradient set of f at $\boldsymbol{x}^*$ is non-empty and $\mathbf{0}$ is a sub-gradient of f at $\boldsymbol{x}^*$, i.e., from the definition of the sub-gradient, $f\,(\boldsymbol{z}) \geq f\,(\boldsymbol{x}^*)$.

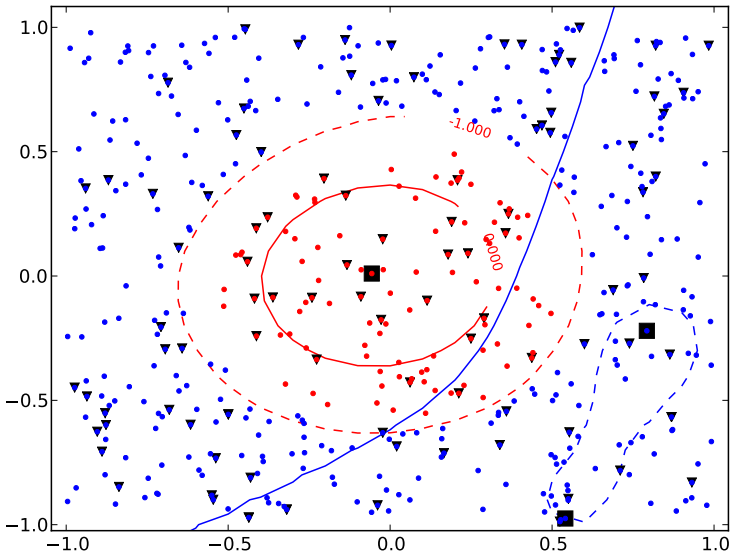


FIGURE 5.3. Impact of pairwise constraints on learning using the stochastic sub-gradient approach.

formed by the SSO-SVM without pairwise constraints (i.e., with $\nu = 0$) is based only on the three training examples and is far from optimal. This clearly shows the correctness of the algorithm and the impact of including pairwise constraints in classification.

5.4. RESULTS

In this section, we present results of our analysis with the new machine learning techniques proposed in this chapter. We use leave-one-complex out cross validation over a set of 70 hetero-dimers selected from the docking benchmark version 4.0 (DBD 4.0). The conditions used for selection of complexes in this subset are:

(1) The number of examples in each complex should be less than 50,000 to limit memory usage. This is required because our implementation of the machine learning models proposed in this section needs the kernel matrices of the training and testing data to fit in memory simultaneously.

(2) The complexes must be hetero-dimeric only because complexes with more units can potentially violate the assumptions behind the pairwise constraints in the proposed models.

The chosen subset of 70 heterodimers is large and diverse enough for a valid performance comparison. However, we first compared the performance of the stochastic sub-gradient optimization (SSO) based SVM (with $\nu = 0$ and $\mu = 0$) to that of the sequential minimal optimization [149] (SMO) based SVM from PyML [150] used in the original PAIRpred on the complete data set of 196 complexes. Table 5.1 presents the leave-one-complex-out cross-validation results for this analysis. Here, we used $K_r = K_{profile} + K_{exp} + K_{HSAAC} + K_{CX}$ as the residue kernel. The value of λ in the SSO based SVM was chosen to correspond to $C = 10$ in the SMO based SVM. Note that both the SVMs give very similar results with the SMO based SVM performing slightly better. However, if the data is centered so that the average of each feature in all training examples is zero, the SSO based SVM gives marginally better RFPP scores. Similar improvements through centering have also been observed for other data sets in [148]. Moreover, the SSO based SVM is, on average, 75% faster than the SMO based SVM. The SSO-SVM (with $\nu = 0$ and $\mu = 0$) can generate predictions for about 50,000 examples in a complex in under 10 minutes using a single processor.

On the reduced data set of 70 complexes, the transductive SVM implementation (algorithm 1) offers better RFPP scores than the original SMO based SVM on the same data set with comparable AUC scores. This shows that transductive SVM offers a small improvement in prediction accuracy. This is especially true for non-rigid complexes in DBD 4.0, i.e., complexes whose constituent proteins undergo a moderate to large conformational change on complex formation (see Table 5.2). For this analysis, $C^\star$ was set to 1.0. However, the

TABLE 5.1. PAIRpred evaluation results with the proposed machine learning models on the complete data set from DBD 4.0 consisting of 196 complexes and the reduced data set of 70 complexes. SMO SVM is the Sequential minimal optimization based SVM used in PAIRpred. SSO SVM is the stochastic sub-gradient optimization based SVM presented earlier. RFPP is the rank of the first positive prediction and AUC is the area under the ROC curve.

Data	Classifier	RFPP (%)					AUC	
		10	25	50	75	90	Complex	Protein
DBD 4.0 (196)	SMO SVM	1	4	20	78	230	85.0	71.4
	SSO SVM	1	5	22	98	319	83.8	70.8
	SSO SVM with centering	1	3	19	83	319	83.3	70.6
DBD 4.0 (70)	SMO SVM	1	2	7	33	154	**84.2**	**71.9**
	SMO based TSVM	**1**	**1**	**5**	**25**	**95**	84.1	71.7
	SMO TSVM with pairwise constraints	1	1	7	40	100	84.1	71.7
	SSO SVM with centering	1	2	6	28	100	82.2	70.7
	SSO SVM centering & pairwise constraints	1	1	6	32	175	80.3	70.9
	SSO centering, pairwise & sparsity constraints	1	3	11	40	139	81.8	70.5

TABLE 5.2. PAIRpred evaluation results with the proposed machine learning models on 22 non-rigid complexes in the set of 70 complexes from DBD 4.0 used in Table 5.1.

Classifier	RFPP (%)					AUC	
	10	25	50	75	90	Complex	Protein
SMO SVM	**1**	3	5	34	192	**81.2**	**69.0**
SMO based TSVM	**1**	**1**	**4**	**28**	203	**81.2**	**69.0**
SSO SVM with centering	1	2	6	20	223	79.0	67.7
SSO SVM with centering & pairwise constraints	**1**	**1**	**5**	**19**	**110**	77.8	**68.9**

transductive SVM is very time consuming to operate. The addition of pairwise constraints does not seem to improve classification performance with the SMO SVM.

The SSO based SVM with $\nu = 0$ and $\mu = 0$ with centering gives marginally better RFPP scores on the reduced data set containing 70 hetero-dimers as well. However, its AUC scores

are lower than the SMO based SVM. Table 5.1 shows that the addition of pairwise constraints to the SSO SVM does not improve the average performance as well. For this analysis, we used a set of 30,000 randomly generated pairwise constraints for each complex with ν set to 0.01[11]. One of the possible reasons for this might be that our classification problem suffers from huge class-imbalance. Pairwise constraints are expected to improve performance when a significant number of constraints have at least one positive example in them. In our case, the number of such constraints is very small because of the class-imbalance. We do observe a minor improvement in RFPP scores and protein level AUC scores with the SSO based SVM with pairwise constraints for non-rigid complexes (see Table 5.2).

We tested the SSO SVM with pairwise constraints with the addition of sparsity constraints for a number of combinations of values of μ, ν, c and ρ. However, the addition of these constraints always resulted in a decrease of performance in comparison to the baseline. Table 5.1 shows a representative result with $\rho = -0.3$, $c = -0.3$, $\nu = 0.001$ and $\mu = 0.001$. Due to the computational complexity of the approach, it is not possible to exhaustively or systematically search for the set of possible parameter values which might improve accuracy relative to the baseline.

[11]Other values of ν generated similar result as well (not shown).

CHAPTER 6

Multiple instance learning of Calmodulin binding sites

In this chapter we focus on prediction of binding sites in proteins that interact with a specific protein: Calmodulin. Focusing on a specific protein allows us to model the special properties of Calmodulin binding sites. Calmodulin (CaM) is an intracellular calcium sensor protein that interacts with a large number of proteins to regulate their biological functions and exhibits sequence conservation across all eukaryotes [151]. Calcium plays a very important role in many cellular functions ranging from fertilization and cellular division to neuronal spiking [152]. Due to the importance of calcium signaling in cells, identifying proteins that bind CaM and determining the location of the CaM binding site in them can help in gaining a better understanding of cellular function in general and the role of calcium in different cellular processes in particular.

We present a highly accurate large margin approach that can identify the location of a CaM binding site in a protein solely on the basis of its amino acid sequence, helping avoid the significant effort of performing such experiments in the lab [152]. We propose a novel algorithm (MI-1 SVM)[12] for binding site prediction and evaluate its performance on a set of 210 CaM-binding proteins extracted from the Calmodulin Target Database [154]. The sites on these proteins that are involved in their interaction with CaM have been annotated through a variety of experimental techniques. However, these annotations are not precise and usually span an area much larger than the true binding site. Thus, not all residues in

[12]Our paper describing MI-1 SVM has been published in a special issue of *Bioinformatics* (2012 impact factor: 5.323) for the European conference on computational biology (ECCB), 2012 [153]. MI-1 SVM is available online at: `http://combi.cs.colostate.edu/supplements/mi1/`.

an annotated binding site interact with CaM. Our prediction scheme directly models the problem of binding site prediction as a large-margin classification problem, and is able to take into account the imprecision in binding site annotations in the training data through multiple instance learning (MIL). Please note that MI-1 SVM, although partner-specific in nature, is very different from PAIRpred in its construction. This is because, MI-1 SVM is geared towards making accurate predictions on proteins that bind Calmodulin using sequence alone. Moreover, the construction of MI-1 SVM has been adapted to handle imprecision in training data annotation.

CaM binding sites are known to be contiguous in sequence, often occurring through an amphiphilic alpha helix [155]. This makes CaM binding site prediction amenable to sliding-window classification approaches [156, 157]. The method by Radivojac et al. [157] uses a hierarchical neural network classifier trained on the basis of amino acid properties averaged over a fixed-size window. Hamilton et al. [156] showed that a simple sliding window SVM trained on average amino acid composition achieves similar performance.

Our novel formulation is based on the framework of multiple instance learning (MIL) [158]. In MIL positive examples come in bags. For a positive bag, it is assumed that at least one of the examples is indeed a positive example. Negative examples are all negative. MIL has been applied in a variety of other problem domains such as object tracking [159], protein identification [160], and prediction of protein-ligand binding affinities [161]. We use MIL for binding site prediction by forming a positive bag out of fixed-size sequence windows that overlap the annotated binding site. This allows us to model the uncertainty in the location of the true binding sites explained earlier. Another advantage of using multiple instance

learning is that it allows us to construct sequence representations that are position dependent, i.e., features that capture both the type of an amino acid and its location in a sequence window simultaneously. This permits learning of motifs that are characteristic of the binding site.

Our results show that the proposed MI-1 SVM has higher accuracy than classical multiple instance SVM (mi-SVM) and is also faster to train. MI-1 also performs better than a standard SVM, thereby improving on existing work of Radivojac et al. [157] and Hamilton et al. [156]. We also compare the merits of several ways of representing binding sites, and demonstrate the ability of our method to learn motifs that are associated with CaM binding. Finally, we show how the resulting binding site predictor can be used as the basis for a classifier that predicts CaM binding proteins, with improved accuracy over earlier work.

6.1. Data Sets and Pre-processing

The data set for CaM binding site prediction and its pre-processing follows the steps given in [157]. A set of 210 proteins was obtained from the Calmodulin Target Database [154]. Each of these proteins bind CaM, and one or more binding sites within each protein are annotated. A non-redundant subset of 153 proteins containing 185 binding sites was then chosen such that no two proteins have more than 40% sequence identity and no two binding sites are more than 50% identical.

Sequence windows of length 21, the average length of CaM binding sites, were extracted from the protein sequences to create positive and negative examples. Negative examples were created by sliding a length 21 window in 10 amino acid increments such that no part of the window overlaps an annotated binding site. Positive examples, on the other hand, were created by sliding a length 21 window over an annotated binding site in increments of

1 amino acid. Thus, the number of positive examples from an annotated binding site equals the number of amino acids in the binding site.

For CaM binding prediction, we used a data set of 241 proteins experimentally determined to bind CaM using a protein array screen that tested around a thousand proteins in *Arabidopsis thaliana* [162]. The remaining 27,138 proteins in the *Arabidopsis thaliana* proteome were used as negative examples (non-binders).

6.2. Classification schemes

Our labeled dataset consists of N labeled examples (x_i, y_i), where x_i is the sequence of a window centered at residue i in a protein, and $y_i \in \{+1, -1\}$ is its associated label indicating whether the central residue of x_i lies in an annotated binding site or not. We denote the feature representation of x_i by $\boldsymbol{\phi}(x_i)$. We used three different classification schemes to predict whether a given residue on a CaM binding protein is involved in binding or not.

6.2.1. Vanilla SVM. First, we experimented with a conventional SVM (referred to as vanilla SVM) [99] given below.

$$\min_{\boldsymbol{w},\boldsymbol{\xi}\geq 0} \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C\sum_{i=1}^{N}\xi_i$$

(13)

$$\text{such that } \forall i: \quad y_i\big(\boldsymbol{w}^T\boldsymbol{\phi}(x_i) + \rho\big) \geq 1 - \xi_i.$$

This is the classifier used in [156]. We used the vanilla SVM to establish the baseline results for different features.

6.2.2. Classical multiple instance learning SVM (mi-SVM). Since not all residues in an annotated binding site may be participating in the interaction with CaM, therefore,

we formulate this problem as a multiple instance learning problem. In this formulation, the positive examples from each binding site are grouped into a single bag. We denote the set of positive examples for a given annotated binding site b as $P(b)$ and the set of negative examples from the protein to which the binding site b belongs as $N(b)$. One possible solution to this problem has been proposed by Andrews et al. as the mi-SVM formulation given below [163].

$$\min_{\boldsymbol{y}\in\{-1,+1\}^N}\left(\min_{\boldsymbol{w},\rho,\boldsymbol{\xi}\geq 0}\frac{1}{2}\boldsymbol{w}^T\boldsymbol{w}+\frac{C}{N}\sum_{i=1}^{N}\xi_i\right)$$

Subject to:

$$y_i\left(\boldsymbol{w}^T\boldsymbol{\phi}(x_i)+\rho\right)\geq 1-\xi_i,\forall i \tag{14}$$

$$\sum_{i\in P(b)}\frac{y_i+1}{2}\geq 1,\forall b$$

$$y_i=-1,\forall i\in N(b),\forall b.$$

In this formulation, the objective is to find the optimal labeling of the examples that comprise the positive bags such that at least one example in each positive bag is labeled as positive. This is mathematically represented by the constraint $\sum_{i\in P(b)}\frac{y_i+1}{2}\geq 1$ in the above formulation. The other constraints ensure correct labeling of the given training examples and that all negative examples are labeled as negative examples. In case of the binding site prediction problem, this means that a trained mi-SVM will choose at least one positive window from the set of positive windows in a binding site. The mi-SVM formulation is a combinatorial optimization problem. We use the heuristic algorithm proposed by Andrews et al. [163] to solve this optimization problem. The algorithm initially assigns the label of a

bag to all examples in it, i.e. all examples in positive bags are assigned a label of +1 whereas all negative examples are assigned −1. It uses these assigned labels to solve a regular SVM learning problem. Labels for all examples in positive bags are then imputed based upon the sign of their discriminant function value. If no example in a positive bag is assigned a positive label (i.e., the constraint $\sum_{i \in P(b)} \frac{y_i + 1}{2} \geq 1$ is violated), the algorithm picks the example in the bag having the largest discriminant function value and sets its label to +1. The algorithm then alternates between label imputation and SVM training until the labels stop changing. This simple algorithm has shown good performance in comparison to more complicated ones [163].

6.2.3. Novel multiple instance learning formulation (MI-1 SVM). Accurate prediction of the location of a binding site in a protein requires a less stringent condition than the one used in mi-SVM: *at least one window in the true binding site needs to score higher than the negative windows from the same protein* (see Figure 6.1). This allows us to significantly reduce the complexity of the learning problem in comparison to mi-SVM. The mi-SVM and vanilla SVM formulations try to classify windows as binding or non-binding without modeling the concept that these windows in fact lie within a protein. Our proposed MI-1 SVM formulation, on the other hand, operates at the protein level. The large-margin formulation of this learning problem, can be expressed as follows:

$$\min_{\boldsymbol{w}, \boldsymbol{\xi} \geq 0} \frac{1}{2} \boldsymbol{w}^T \boldsymbol{w} + \frac{C}{M} \sum_b \xi_b$$

(15) such that $\forall b$:

$$max_{i \in P(b)} \left(\boldsymbol{w}^T \boldsymbol{\phi}\left(x_i\right) \right) \geq \boldsymbol{w}^T \boldsymbol{\phi}\left(x_j\right) + 1 - \xi_b, \forall j \in N\left(b\right).$$

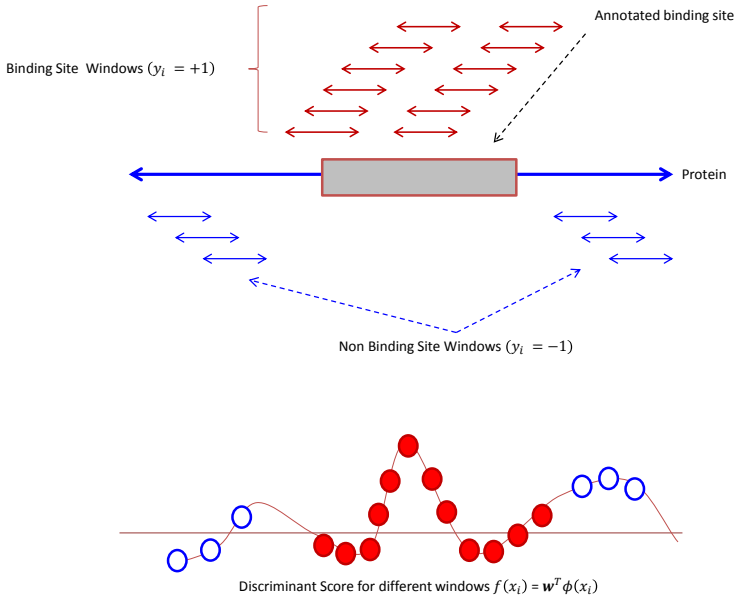


FIGURE 6.1. The CaM binding site prediction problem. The annotated binding site is shown as a box, and is represented by a bag composed of the windows indicated in red above the binding site. The rest of the windows that do not overlap the binding site are negative examples (shown in blue below the protein). The bottom panel illustrates the desired characteristics of the classifier's discriminant function. The dots indicate the score of different examples (positive indicated by solid red circles and negative shown as hollowed blue circles). The score from the trained discriminant function for one window in a binding site should be higher than the scores generated for non-binding site windows within that protein.

Here M is the total number of binding sites in the training data. For a given binding site, this formulation tries to maximize the difference between the discriminant function values of the maximum scoring window within the binding site and the non-binding windows in the rest of the protein containing the binding site. MI-1 SVM formulation of the problem offers a number of advantages over vanilla SVM and mi-SVM. MI-1 does not require a bias term since it simply compares the discriminant function scores of the binding and non-binding windows in its constraints. Also, the number of slack variables (ξ_b) in MI-1 SVM is equal to the number of binding sites and not the number of training examples, as in the vanilla SVM and the mi-SVM formulations. As a consequence, the number of variables involved in

the optimization in MI-1 SVM is much smaller than that in mi-SVM and this leads to faster training. Using the same ξ_b for a single binding site effectively takes the maximum of the scores over all non-binding site windows of the protein to which the binding-site b belongs. Another important feature of MI-1 SVM is that, like ranking-SVM [164], MI-1 SVM also explicitly maximizes the area under the Receiver Operating Characteristics (ROC) curve.

Similar to mi-SVM which performs optimization over the labels of examples in positive bags, MI-1 SVM is also a combinatorial optimization problem because of the maximum operation in its constraints. We have used the heuristic algorithm given below to obtain a solution to this problem. The algorithm can be stopped when the representative examples of all binding sites stop changing, or on the basis of a user-defined maximum number of iterations. In all our experiments, the algorithm converged in 10 iterations or less. A trained MI-1 SVM can be used to produce discriminant function scores for any given residue in a protein.

Algorithm 3 MI-1 SVM

1: Initialization: With each binding site b, associate a representative example x^b, with feature representation $\boldsymbol{\phi}\left(x^b\right)$ which is initialized to the mean of the examples in $P\left(x^b\right)$:

$$\boldsymbol{\phi}\left(x^b\right) = \frac{1}{\mid P\left(x^b\right) \mid} \sum_{i \in P\left(x^b\right)} \boldsymbol{\phi}\left(x^b\right), \forall b$$

2: **repeat**

3: Solve the following quadratic programming optimization problem

$$\min_{\boldsymbol{w}, \boldsymbol{\xi} \geq 0} \frac{1}{2} \boldsymbol{w}^T \boldsymbol{w} + \frac{C}{M} \sum_b \xi_b$$

Subject to $\forall b$:

$$max_{i \in P(b)} \left(\boldsymbol{w}^T \boldsymbol{\phi}\left(x^b\right)\right) \geq \boldsymbol{w}^T \boldsymbol{\phi}\left(x_j\right) + 1 - \xi_b, \forall j \in N\left(b\right).$$

4: Update $(\forall b)$: $\boldsymbol{\phi}\left(x^b\right) = \boldsymbol{\phi}\left(x_i\right)$, such that $i = argmax_{j \in P(b)} \boldsymbol{w}^T \boldsymbol{\phi}\left(x_j\right)$.

5: **until** convergence conditions are met.

The quadratic programming problem in the MI-1 algorithm can be solved in the primal or in the dual. The primal formulation of the problem is more efficient than the dual when the dimensionality of the feature vector is smaller than the number of training examples. The dual formulation of the quadratic programming problem (based upon the Lagrangian of the primal objective function) is given by:

$$\min_{\alpha} \left(\frac{1}{2} \sum_{b,j \in N(b)} \sum_{c,i \in N(c)} \alpha_i^c \left(\Delta_i^{cT} \Delta_j^b \right) \alpha_j^b - \sum_{b,j \in N(b)} \alpha_j^b \right)$$

such that $\forall b$:

(16)

$$\alpha_j^b \geq 0, j \in N(b)$$

$$\sum_{b,j \in N(b)} \alpha_j^b \leq \frac{C}{M}.$$

Here α_j^b is the Lagrange variable corresponding to the primal constraint $\boldsymbol{w}^T \boldsymbol{\phi}\left(x^b\right) \geq \boldsymbol{w}^T \boldsymbol{\phi}\left(x_j\right) + 1 - \xi_b$ and $\Delta_j^b = \phi\left(x^b\right) - \phi\left(x_j\right)$. The dual formulation reveals some interesting aspects of the MI-1 SVM. It shows that Lagrange variables α only exist for negative examples, and that the sum of all α for negative examples from a single protein is constrained to be less than or equal to $\frac{C}{M}$. This differs from a conventional SVM formulation which requires that each of the α, on its own, should be less than or equal to $\frac{C}{M}$ and the sum of products of α from all training examples with their corresponding labels should be zero. Thus, the MI-1 SVM formulation is less constrained than a conventional SVM formulation and this can, potentially, lead to achieving a better solution.

6.3. CaM Binding Prediction

In this work, we compare the following two strategies for predicting whether a protein binds Calmodulin or not.

6.3.1. Discriminant Function Scoring. The maximum discriminant function score across all windows in a protein can be used as the CaM binding propensity of that protein. This approach was used in [156] to predict CaM binding of proteins in the *Arabidopsis thaliana* proteome. In their method, the scores were generated using a standard SVM classifier trained for binding site prediction. In this paper, we use the scores from MI-1 SVM instead, i.e., the maximum value of the discriminant function score from a trained MI-1 SVM across all windows in a protein is taken as the propensity with which that protein is expected to bind CaM.

6.3.2. Cascaded Classification. We implemented a two stage cascaded classification approach for CaM binding prediction. In the first stage the window in a given protein with the highest MI-1 SVM discriminant function score is chosen as the most likely binding site window for that protein. This is done for all proteins in the training set. In the second stage, a standard SVM is trained to discriminate between the most likely binding site windows in positive examples (known CaM binding proteins) and negative examples (non CaM-binding proteins). Once the second stage SVM has been trained, the binding propensity of a test protein can be estimated by first finding its most likely binding site window using MI-1 SVM, and then evaluating the discriminant function value of the second stage SVM for the chosen window. A Gaussian kernel was used in the second stage SVM as it performed significantly better than a linear kernel. However, the use of non-linear kernels in MI-1 SVM did not seem to improve performance.

The performance of the learning methods described above for binding site prediction was analyzed using the following sequence based feature representations.

p-spectrum: The p-spectrum of a string over an alphabet $\mathbf{\Sigma}$ is a vector $\boldsymbol{\phi}(x)$, each of whose components $\boldsymbol{\phi}_v(x)$ is the number of occurrences of each length-p substring v in the string x. For protein sequences, $\mathbf{\Sigma}$ is the set of the 20 amino acids. The p-spectrum kernel between two strings is given by the Euclidean dot product of their p-spectra [165].

Position dependent p-spectrum: For a given string x, the position dependent p-spectrum $\boldsymbol{\phi}(x)$ is represented by a vector of indicator variables $\boldsymbol{\phi}_{v,k}(x)$ each showing whether the string x contains the length-p substring v at position k in the string x or not. The position dependent p-spectrum kernel $K^{PD}(x,z) = \boldsymbol{\phi}(x)^T \boldsymbol{\phi}(z)$ is the number of common substrings occurring at the same locations in the two strings x and z. The position dependent kernel takes the relative position of an amino acid in a window into account whereas the p-spectrum kernel does not.

Position dependent gappy triplet: This feature representation quantifies the occurrences of motifs of the form $a \times^m b \times^n c$, where a, b, c are amino acids and $\times^m$ indicates m don't-care positions. For a given string x, the feature vector $\boldsymbol{\phi}^{m,n}(x)$ of the position dependent gappy triplet comprises of variables $\boldsymbol{\phi}^{m,n}_{a,b,c,k}(x)$ which indicate whether the motif $a \times^m b \times^n c$ starts at position k in the string or not. The kernel $K^{m,n}(x,z) = \boldsymbol{\phi}^{m,n}(x)^T \boldsymbol{\phi}^{m,n}(z)$ between two strings tells us the number of locations in the two strings that have the same motif starting at them. We have used multiple position dependent gappy triplet kernels as $K^{PDGT}(x,z) = \sum_{m=0}^{4} \sum_{n=0}^{4} K^{m,n}(x,z)$.

This kernel allows us to extract meaningful information about motifs for CaM binding sites and is only used for binding site prediction for this purpose.

We normalize any kernel using the cosine kernel $K_{cos}(x,z) = \frac{K(x,z)}{\sqrt{K(x,x)\,K(z,z)}}$ which corresponds to normalized each feature vector $\boldsymbol{\phi}(x)$ to have unit norm.

6.5. Performance Evaluation

We use Leave-One-Protein-Out (LOPO) cross validation in order to analyze the performance of the techniques described above for binding site prediction. In LOPO cross validation, all examples (positive or negative) from a single protein are held-out while the classifier is trained on the remaining proteins. The classifier is then evaluated over the examples from the held out protein. We evaluate the following performance metrics and use their average across all proteins to make comparisons between methods and kernels:

Protein level area under the ROC curve (AUC): The area (expressed as percentage) under the Receiver Operating Characteristic (ROC) curve (the plot of true positive rate versus false positive rate) obtained for windows in a given protein.

Protein level Area under ROC 10% Curve ($\mathbf{AUC_{0.1}}$): The area (expressed as percentage) under the ROC curve based on up to the first 10% false positives in a protein.

False Hit Ratio (FH-measure): The percentage of non-binding site windows (out of the total number of non-binding site windows) that have a score higher than the maximum scoring window in the known binding site. This measure tells us how many non-binding site windows are expected with a score higher than the true binding site window.

True Hit Probability (TH-measure): For a given protein, a true hit is defined to occur when the residue at the center of the highest scoring window for that protein lies within a binding site. The average number of true hits across all proteins (called the TH-measure) represents the probability of the maximum scoring window predicted by a classifier to lie within a true binding site.

In the context of this problem, the AUC is a measure of how good a particular method is in ranking binding site windows above non binding sites. $AUC_{0.1}$ gives us a sense of how good are the top scoring windows produced by a classifier. The FH measure represents the chances of a non-binding site window to be ranked higher than a true binding site window. The TH-measure tells us about the chances of the highest scoring window predicted by a classifier to belong to a true binding site. Both the TH and the FH measures provide very meaningful information about the accuracy of the method to a biologist who intends to use the proposed prediction scheme to verify potential binding site locations experimentally.

We use AUC as the performance metric for CaM binding prediction. AUC can be directly computed from the estimated CaM binding propensities when using the Discriminant function scoring approach. With the Cascaded classification approach, AUC is obtained from 5-fold stratified cross-validation with nested grid search for model selection. In cross-validation, it was ascertained that two proteins with more than 40% sequence similarity are in the same fold (evaluated using BLASTCLUST from the NCBI BLAST package [166]). Moreover, the data for CaM binding prediction in A.thaliana did not include any proteins which were part of the MI-1 training set.

6.6. Model Selection

In order to perform model selection (the choice of the cost parameter C) for the vanilla and MI-1 SVM formulations for binding site prediction, we have used nested 5-fold cross validation within each iteration of the LOPO cross validation process. The TH-measure obtained from the 5-fold cross validation is then used to choose the value of C for that iteration of LOPO cross validation. The values of C that were used in the nested cross validation are $\{0.01, 0.1, 1.0, 10, 100\}$.

As mi-SVM takes a long time to train, nested cross validation could not be performed. Instead we evaluated the LOPO cross validation performance (TH-measure) of mi-SVM with different values of C in $\{0.01, 0.1, 1.0, 10, 100\}$ and the best results with the optimal value of $C = 10$ are reported. It should be noted that this method for selection of C for mi-SVM can potentially lead to over optimistic performance estimates. This is not an issue, since our claim is that the proposed approach performs better.

The window size was chosen as 21 as it is the average number of amino acids in known CaM binding sites in our dataset.

In the case of CaM binding prediction in proteins from *Arabidopsis thaliana* using cascaded classification, we performed a nested (5-fold) grid search within each cross validation fold for selecting the parameter values of the second-stage SVM. In the grid search, optimal values of C in the SVM and γ of the Gaussian kernel $K(x_1, x_2) = exp(-\gamma|x_1 - x_2|^2)$ were chosen from $\{0.1, 1, 10, 100\}$ and $\{0.005, 0.02, 0.5, 2.0\}$ respectively. The data for CaM binding prediction in *A. thaliana* did not include any proteins which were part of the MI-1 training set.

6.7. Results

Table 6.1 presents the leave-one-protein-out cross validation results for the three SVM formulations for the 1-spectrum, position dependent 1-spectrum and the combination of the two feature representations for predicting CaM binding sites. We observe that both MIL formulations (mi-SVM and MI-1 SVM) perform better than the vanilla SVM. This shows the value of expressing binding site prediction as a multiple instance learning problem. This is particularly evident with the use of position dependent feature representations, as they are more sensitive to changes in relative position of an amino acid in a window within the binding site than position independent feature representations. It can also be noted that the accuracy of MI-1 SVM is noticeably better than mi-SVM. We believe that this improvement stems from the fact that the proposed scheme implements a more realistic model of the binding site prediction problem. The improvement resulting from switching to a position dependent feature representation is also larger for MI-1 SVM than that observed in the case of mi-SVM. The higher $AUC_{0.1}$ scores indicate the improved sensitivity and specificity of MI-1 SVM which is also reflected in the 8% improvement in the TH-measures and the decrease in the FH-measure.

The vanilla SVM approach is the same as the method in [156], which they showed works comparably as the neural network approach of [157]. Therefore we conclude that the proposed scheme performs better than previously reported approaches.

We also compare the performance of these approaches with a naive local alignment based method for finding CaM binding sites. In this method, local alignment between a held out protein and the binding sites of the remaining proteins is performed and if the best scoring alignment overlaps (by at least ten residues) with the known binding site in the held out

protein, it is considered to be a true hit. This approach gives a TH% of 39.5%. This shows that the machine learning approaches presented in this paper use more than mere sequence identity to make better predictions.

We have also performed an analysis of the stability of the results for the MI-1 and the Vanilla SVMs by averaging performance statistics of 12 runs of 5-fold cross validation. This analysis was not performed for the mi-1 SVM owing to its large time requirements. The 5-fold cross validation results for both the methods are very similar to the LOPO cross validation results. The maximum standard deviation in a particular performance metric across different feature representations obtained from the 5-fold cross validation is given in Table 6.1. This statistic gives an idea of the variability of the results with respect to changes in the data.

Figure 6.2 shows the output of the MI-1 SVM for a single protein for the position dependent and position independent versions of the 1-spectrum feature representation. It should be noted that out of the two binding sites in the protein, only one is predicted correctly

TABLE 6.1. Results across different methods and feature representations (kernels). Here, Max. Std. indicates the maximum standard deviation of different performance metrics for different methods obtained through 5-fold leave protein out cross validation. These values could not be obtained for mi-SVM because of its long training times.

Method	Features	AUC	$AUC_{0.1}$	TH %	FH %
Vanilla SVM	1-Spec	95.5	53.9	**66**	2.6
	PD-1	95.6	54.5	64	2.5
	Comb.	95.9	55.1	65	2.1
	Max. Std.	0.16	0.59	2.2	0.15
mi-SVM	1-Spec	95.5	**54.4**	64	2.6
	PD-1	96.0	55.8	69	2.1
	Comb.	96.2	55.6	68	1.9
MI-1 SVM	1-Spec	**96.0**	54.3	62	**2.1**
	PD-1	**96.8**	**58.5**	**72**	**1.3**
	Comb.	**96.9**	**59.0**	**75**	**1.2**
	Max Std.	0.14	0.80	3.4	0.11

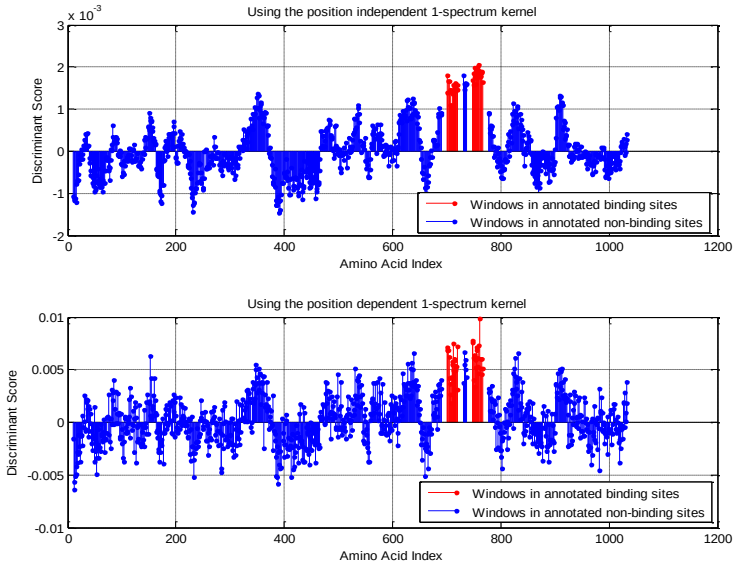


FIGURE 6.2. MI-1 discriminant values along the length of a held-out protein with the position independent (top) and the position dependent (bottom) 1-spectrum features.

(i.e., a window in the binding site has a score higher than all non-binding site windows) by the position independent features, whereas the position dependent features are able to predict both annotated binding sites accurately. This clearly illustrates the usefulness of position dependent feature representation. It is quite clear that the output for the position-independent features is much smoother than that from the position-dependent 1-spectrum features. This is because the position-independent 1-spectrum feature vector representation changes only slightly as the window is translated by one position, whereas the position-dependent feature vector can change dramatically. Due to the increased resolution power, the position-dependent features lead to a classifier that is able to correctly predict both binding sites in the example shown in Figure 2, which is not achieved using the position-dependent features.

On the task of CaM binding prediction (see Table 6.2), the performance of discriminant function scoring is only marginally better than that of the 1-spectrum feature representation

used in [156]. However, with the cascaded classification approach with a Gaussian kernel, the results are significantly better. Even though the AUC for the position-independent 1-spectrum features is higher than that of the position-dependent features, the $AUC_{0.1}$ was higher for position-dependent features (29.1) in comparison to the simple 1-spectrum features (26.6).

In order to obtain a better understanding of what our classifier picks up, we considered the proteins that are not known to bind to CaM and ranked that list according to the score provided by our classifier. We then tested for enrichment of GO terms of segments of that list: the first 1000 proteins, proteins 1001-2000 etc., using the GOrilla tool [167]. For the first 1000 we found enriched terms that are in agreement with known functions of CaM binders [152]: In GO molecular function, transcription function activity and CaM-dependent kinase activities were the most highly enriched with adjusted p-values below 10^{-10}. All other enriched terms were related to these except for 'inward rectifier potassium channel activity' which had an adjusted p-value of 0.02. In GO biological process namespace all the terms except for 'response to carbohydrate stimulus' (adjusted p-value 0.02) were related to phosphorylation and various regulatory processes. In analyzing enrichment for size-1000

TABLE 6.2. The features are 1-spectrum (1-Spec), position-dependent 1-spectrum (PD-1) and the combination (Comb) of the 1-Spec and PD-1 feature representations. Using Cascaded Classification with a liner kernel in the second stage SVM instead of the Gaussian kernel, the best AUC was 0.72 with 1-spectrum features. (AUC: area under the ROC curve).

Method	Features	AUC
Discriminant function scoring	1-Spec	71.9
	PD-1	70.1
	Comb.	71.9
Cascaded classification	1-Spec	**75.3**
	PD-1	**71.1**
	Comb.	**72.3**

chunks we found that the p-values for these functions and processes went down as we went down the ranked list, and for proteins ranked 5000-6000, no terms showed enrichment.

6.8. Recovering binding motifs

We analyzed the weights from different features in order to extract amino-acid composition related information for CaM binding sites. The plot of weights from the 1-spectrum features and the position dependent 1-spectrum features are shown in Figure 6.3. The weights for the 1-spectrum features closely follow the amino acid propensities in CaM bindings sites as observed in [156], with R (Arginine), K (Lysine) and W (Tryptophan) showing large positive weights, whereas D (Aspartic acid), E (Glutamic acid) and P (Proline) have large negative weights. The plot of the position dependent 1-spectrum features indicates that the importance of different amino acids changes with their position in the window. For example, Arginine shows large positive weights in the middle of the window, and negative weights in the ends; Glutamic acid shows the opposite behavior. This indicates that the classifier is indeed learning a position dependent model.

The results of 5-fold cross validation using the position dependent gappy triplet kernel K^{PDGT} shown in Table 6.3 indicate that this kernel provides comparable performance to other feature representations using MI-1 SVM. Since the number of dimensions in the feature representation of the gappy triplet kernel is much larger than the number of training examples, MI-1 SVM learning was performed using the dual formulation for this kernel, which is more computationally intensive. That is why we have used 5-fold cross validation instead of LOPO cross-validation.

Next, we ranked the features of the gappy triplet kernel in terms of their weights in MI-1 SVM learning in order to find motifs that are associated with CaM binding. Figure 6.3

shows the top 100 motifs and their positions. We observe that motifs tend to associate with particular positions, showing that MI-1 SVM uses the flexibility in choosing a representative window to 'align' instances of CaM binding sites (for instance, notice the presence of 'R' at positions 10 and 11). Moreover, it is able to find parts of known CaM binding motifs provided in the CaM Target Database [154]. The CaM Target Database classifies CaM binding targets into 5 groups, each characterized by certain motifs: three predominantly calcium dependent motifs (1-10, 1-14 and 1-16, named according to the position of large hydrophobic residues), the IQ motif which is typically not dependent on calcium concentration, and others. As is evident from Figure 6.3, IQ, QxxxR, RxxxxR, RGxxxR, RxxL, KxxxxR receive large positive weights. These motifs are components of the IQ subclass of motifs. Other features belonging to different subclasses of motifs that receive large positive weights include: AxxI, IxxxF, LxxV, (from the 1-14 subclass), RR, KK, RxF (from the 1-10 subclass) etc. This clearly illustrates the capabilities of the proposed scheme to learn CaM binding motifs. We also note that most of the top ranking features correspond to a motif with 3 or 4 don't care positions. This is in agreement with the known fact that CaM binding usually occurs via an alpha helix, and this corresponds to the periodicity of the alpha helix.

6.9. MI-2: Simultaneous prediction of binding and binding sites

We also developed a large-margin formulation (called MI-2) to simultaneously learn whether a protein binds CaM or not and its potential binding site. The intuition behind the development of MI-2 is that the two tasks (binding site prediction and binding prediction)

Table 6.3. Results for the position dependent gappy triplet kernel.

Method	*AUC*	$AUC_{0.1}$	*TH %*	*FH %*
MI-1 SVM	96.5	58.5	68	1.6

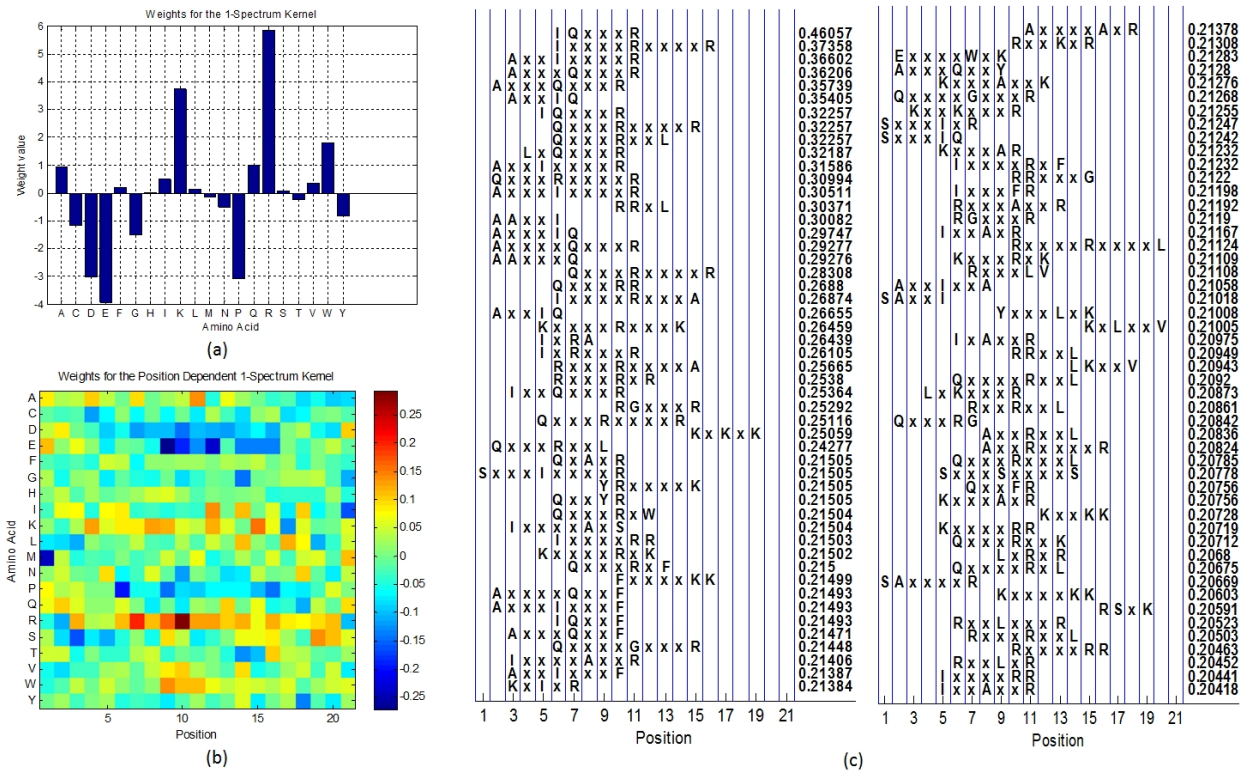


FIGURE 6.3. (a) Weights of different amino acids in the (position independent) 1-spectrum feature representation (b) Heat map of the weights of different amino acids versus their position from the MI-1 SVM position dependent 1-spectrum feature representation and (c) Top 100 (in terms of their weights) motifs from the position dependent gappy triplet kernel. The last (numeric) column shows actual weight values.

are very closely related and the joint learning of the two tasks can potentially improve prediction accuracy on both tasks. In this section, we first describe this formulation and then present its results.

Let B be the set of all binding sites from the set of 153 proteins for which the location of the binding site is known. As earlier, for any binding site $b \in B$, the set of binding and non-binding windows in the protein containing b are denoted by $P(b)$ and $N(b)$, respectively. B^+ denotes the set of the known 241 CaM binding proteins in *A. thaliana*. B^- is the set of

remaining proteins in *A. thaliana*. The formulation can then be written as follows:

$$\min_{\boldsymbol{w},\rho,\boldsymbol{\xi},\boldsymbol{\xi}^+,\boldsymbol{\xi}^-} \left(\frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + \frac{C}{\mid B \mid} \sum_{\forall b \in B} \boldsymbol{\xi}_b + \frac{C^+}{\mid B^+ \mid} \sum_{\forall p \in B^+} \boldsymbol{\xi}_p^+ + \frac{C^-}{\mid B^- \mid} \sum_{\forall p \in B^-} \boldsymbol{\xi}_p^- \right)$$

Binding site constraints $\forall b \in B$:

$$\forall j \in N(b) \; : \; \max_{i \in P(b)} \left(\boldsymbol{w}^T \boldsymbol{\phi}(x_i) \right) \geq \boldsymbol{w}^T \boldsymbol{\phi}(x_j) + 1 - \xi_b$$

$$\xi_b \geq 0$$

Binding prediction constraints:

$$\forall p \in B^+ \text{ and } \forall i \in p : \; \boldsymbol{w}^T \boldsymbol{\phi}(x_i) + \rho \geq 1 - \xi_p^+, \boldsymbol{\xi}_p^+ \geq 0$$

$$\forall p \in B^- \text{ and } \forall i \in p : \; \boldsymbol{w}^T \boldsymbol{\phi}(x_i) + \rho \leq -1 + \xi_p^-, \boldsymbol{\xi}_p^- \geq 0.$$

The objective of the above formulation is to learn an objective function $f(x) = \boldsymbol{w}^T \boldsymbol{\phi}(x) + \rho$ that can predict whether the input sequence window x is part of a binding site or not. The maximum discriminant function score of a protein can be used to predict whether it can bind CaM or not. Note that the binding site constraints in MI-2 are identical to those in MI-1. The binding prediction constraints in MI-2 ensure that the maximum of the discriminant function scores of all windows in a CaM binding protein is higher than $+1$ and that it is lower than -1 for non-binding proteins. Note that the binding prediction constraints use a single slack variable ξ_p^+ or ξ_p^- for all examples from a protein. This implicitly applies the maximum operation over all examples from the same protein and enforces the constraints discussed earlier without introducing any non-linearities in the formulation. The parameters C, C^+ and C^- control the relative contribution of the three sets of constraints. Since B^- can potentially contain CaM binders as well, C^- is taken to be very small in comparison to

C and C^+. This formulation can be solved with straightforward additions to the heuristic algorithm for MI-1 to incorporate the changes in the objective function and constraints in the quadratic programming problem.

In order to evaluate the performance of MI-2 for binding site prediction, 5-fold cross validation was performed on the data set of 153 proteins for which the binding sites are known. The training set for binding prediction was kept the same across all cross-validation folds. Using the combination of position dependent and independent 1-spectrum features, the results for binding site prediction with MI-2 are as follows: $AUC = 95.3$, $AUC_{0.1} = 56.3$, $TH\% = 71.3$ and $FH\% = 1.3$. These results are marginally inferior to the results from MI-1. However, they are better than the vanilla SVM and classical mi-SVM formulations.

For the task of binding prediction, we performed 5-fold cross validation on $B^+ \cup B^-$. The binding site prediction data set was not changed during these cross validation folds. This gave an AUC of 74.0 which is slightly better than the results from discriminant function scoring with MI-1 but slightly lower than the results from cascaded classification.

6.10. Discussion

We have presented a novel MIL algorithm for CaM binding site prediction called MI-1 SVM, and shown its performance advantages in comparison to the standard MIL SVM and regular SVM, which was used in previous work. Our new MIL formulation captures the minimal constraints that a good binding site classifier needs to have, and we believe this is the reason for its better accuracy. Not only that, it also runs more than twice as fast as standard MIL SVM (running time on a dataset of 16,060 windows was 510.5s for MI-1, 1059.1s for mi-SVM, and 348.3s for vanilla SVM). Expressing binding site prediction as an MIL problem is a natural way to incorporate uncertainty about binding site location, and our results show

that this allows the classifier to 'align' binding sites and learn position-dependent motifs that characterize the binding site. The proposed scheme also shows its efficacy in prediction of CaM binding proteins. In addition, we also analyzed the performance of a model (MI-2) for the joint learning of binding sites and CaM binding. However, we could not attain major performance improvements using this model in comparison to MI-1.

CHAPTER 7

Conclusions and Future work

In this work we have proposed a number of large-margin models for the prediction of binding sites and interfaces. Specifically, we presented a generic predictor called PAIRpred which explicitly models the partner-specific nature of protein complex formation using a pairwise support vector machine. PAIRpred can predict the binding sites in individual proteins in a complex as well as inter-residue contacts between two proteins using sequence alone or in conjunction with structural features. PAIRpred offers state of the art accuracy for both interface and binding site prediction. We also presented a multiple instance learning based approach for predicting the binding sites in proteins that bind Calmodulin using only sequence. Major contributions to the field from this work can be enumerated as follows:

Impact of partner-specific predictions: PAIRpred is the first method to make partner-specific predictions of protein interfaces from structure. PAIRpred offers better accuracy for binding site prediction at the protein level in comparison to an SVM classifier (AUC of 77.0 compared to 72.6) trained using the same set of features over the same evaluation data set. This clearly shows that modeling the partner-specific nature of protein complexes can lead to better predictions. Moreover, using PAIRpred we were able to replicate, *in silico*, an experimental study for analyzing the binding specificity of the NS1B protein from the Influenza A virus. Such an analysis is possible only with partner-specific predictors like PAIRpred.

Accuracy of interface prediction: PAIRpred currently offers the best accuracy for both binding site prediction at the protein level and interface prediction at the complex level. We attribute this accuracy to the use of a pairwise SVM for classification.

Impact of different types of features: We have found that sequence and structural conservation through local alignments, surface exposure, local surface geometry and template based features contribute to the accuracy of prediction. The pairwise SVM in PAIRpred allows us to extract features at the residue level and combine them with pairwise features in a seamless manner through its use of kernels.

Transductive and semi-supervised approaches: We have investigated the design of transductive and semi-supervised machine learning schemes to better model the nature of the problem. We have found that transductive models can provide limited improvement in prediction accuracy. However, our attempts at modeling pairwise labeling constraints between classification examples from a query complex and the sparsity of the prediction did not contribute towards improvement in accuracy.

Impact of conformational change on accuracy: We have found that binding associated conformational change in proteins is one of the major factors that limit accuracy of interface and binding site prediction. PAIRpred offers a more graceful degradation of prediction performance with respect to the degree of conformational change in comparison to other existing methods.

Novel multiple instance learning models: We have also developed two large-margin multiple instance learning models which are able to handle imprecision in binding site annotations. These models achieve state of the art accuracy in binding site prediction for Calmodulin binding proteins.

7.1. Future Work

As future work, a list of some more ideas in relation to binding site prediction is presented below.

7.1.1. Incorporation into docking methods. Output from our partner-specific predictor can be used to improve docking solutions either by re-ranking the models produced by a docking method [168, 75] or incorporating the predicted residue pair binding propensities directly into the energy function used in such methods [169, 76].

7.1.2. Handling binding associated conformational changes. One of the biggest challenges in predicting binding sites or protein interfaces are the conformational changes resulting from binding. This issue is compounded by the fact that complexes with large conformational changes are under-represented in the PDB. Most existing machine learning methods for interface prediction do not explicitly model protein flexibility or incorporate features based on protein flexibility. We would greatly encourage future researchers to focus on this aspect of the problem. One possibility can be to extract features from the NMR structures of proteins and complexes, as such structures do capture the dynamics and flexibility of proteins. However, currently the number of complexes with NMR structures is very small. A search for protein NMR structures with two or more chains in their biological units results in approximately 1200 hits whereas this number is almost 50,000 for X-ray structures.

7.1.3. Using protein-protein interaction network data. A number of high-throughput experimental techniques can detect existence of protein-protein interactions (PPIs) but they do not provide any specific information about the binding interface between these proteins. An interesting problem is the prediction of protein binding regions from a protein-protein interaction network. Some computational methods [85, 177], predict protein domain interactions given such a network. However, to the knowledge of this author, no computational prediction scheme for finding binding sites at the residue level

from protein-protein interaction data exists in the literature. Here, a brief description of the problem is given and a possible solution through multiple instance learning is described.

Assume that we have been given a protein interaction network in which each node is a protein and the edges indicate the existence of an interaction between two proteins. Specifically, let U denote the set of pairs (A, B) of proteins in the given network so that y^{AB} indicates whether these proteins interact ($y^{AB} = +1$) or not ($y^{AB} = -1$). The objective here is to find out whether a pair of residues (denoted by i) in a protein complex interact ($\overline{y}_i^{AB} = +1$) or not ($\overline{y}_i^{AB} = -1$). Let's assume further that we also know the interfaces for a set S of protein complexes. Formally, given a residue pair i in the complex (A, B) in S, we can use y_i^{AB} to indicate whether this pair is known to interact ($y_i^{AB} = +1$) or not ($y_i^{AB} = -1$). We can use the following large-margin formulation to generate a predictor for inferring whether a residue pair in a query complex is likely to interact or not. It can also be used for finding the interfaces on complexes in the training data.

$$\min_{\boldsymbol{w},b} \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C_U \sum \xi^{AB} + C_S \sum \xi_i^{AB} \tag{17}$$

Subject to:

$$y^{AB} \max_{i \in AB}(\boldsymbol{w}^T \boldsymbol{x}_i^{AB} + b) \geq 1 - \xi^{AB}, \xi^{AB} \geq 0, \qquad \forall (A, B) \in U \tag{18}$$

$$y_i^{AB}(\boldsymbol{w}^T \boldsymbol{x}_i^{AB} + b) \geq 1 - \xi_i^{AB}, \xi_i^{AB} \geq 0, \qquad \forall i \in AB, \forall (A, B) \in S. \tag{19}$$

The set of constraints in Equation (18) state that no residue pair in a non-interacting pair of proteins should be classified as interacting and at least one residue pair in a known protein complex should be classified as interacting. Thus, these constraints ensure that the classifier *explains* the interaction network properly. The parameter C_U in Equation (17)

controls the cost of misclassification of edges in the interaction network. The constraints in Equation (19) ensure correct classification in cases with known binding interfaces. Once the above optimization problem has been solved, $\overline{y}_i^{AB} = \boldsymbol{w}^T \boldsymbol{x}_i^{AB} + b$ can be used to rank pairs of interacting residues in training complexes for which the binding interfaces were not previously known. The above model seamlessly integrates interaction and interaction site data for training a predictor which can potentially be more accurate than a predictor that uses only interaction site data. This formulation can be viewed as a generalization of the multiple instance learning based MI-1 SVM described earlier [153].

7.1.3.1. *Prediction of binding sites in Flaviviridae proteins.* A special case of the problem described in the previous subsection is the interaction network between the NS3 and NS5 proteins from various Flaviviridae viruses. Flaviviridae include, among others, Dengue, West Nile, Hepatitis and Yellow Fever viruses. The NS3 and NS5 proteins in these viruses are thought to play a vital role in replication of these viruses [178, 179, 180]. There is significant evidence that the NS3 protein from one virus species interacts with only the NS5 protein from the same species [181]. That is, NS3 protein from Dengue virus does not interact with that from the West Nile virus and so on. Figure 7.1 shows the protein-protein interaction network for these viruses which points out the species-specific nature of the interaction of these proteins. Thus, for the PPI network of Figure 7.1, we do know which proteins interact but do not know the residues involved in these interactions. The NS3 proteins from different Flaviviridae species have a large number of conserved residues across the species. The same holds true for NS5 proteins. Therefore, the residues which differ between proteins from different species hold the key to explain the species-specific pattern of interactions.

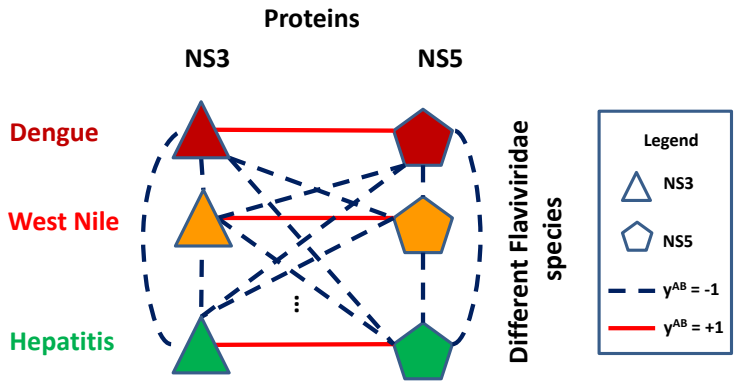


FIGURE 7.1. The special protein-protein interaction network for NS3 and NS5 proteins in Flaviviridae.

This information can be modeled by the large-margin formulation proposed in the previous subsection to locate the residues involved in the interaction for individual proteins.

7.1.4. SPECTRAL FEATURES. Proteins sequences can be treated as multidimensional, non-stationary and non-linear signals. Different properties of a protein sequence (such as hydrophobicity and accessible surface area) can be taken as components of these multidimensional signals [182]. The frequency characteristics of these signals can change over the length of the sequence giving them a non-stationary nature. Moreover, sequence properties are a result of complex and usually non-linear thermodynamic and evolutionary laws that govern protein folding and binding. As a consequence, protein sequences can be subjected to spectral analysis techniques such as wavelets and the Hilbert-Huang transform [183, 184, 185] in order to extract spectral features to study the spectral properties of protein sequences and their relation to binding sites. To the knowledge of this author, no existing method has used such features for predicting binding sites. Spectral features derived from protein structures can be very helpful in modeling the surface characteristics of proteins and can result in more accurate interface predictions. However, extracting such features requires significant insight in the working of these methods and devising extensions of such methods that can operate

on a multidimensional feature representation of physiochemical and structural properties of protein surfaces.

Bibliography

[1] A. Kessel and N. Ben-Tal, *Introduction to Proteins: Structure, Function, and Motion*. Taylor & Francis US, Dec. 2010.

[2] R. A. Freitas Jr., "Human body chemical composition (section 3.1)," in *Nanomedicine: Basic Capabilities*, vol. 1, Landes Bioscience, 1999.

[3] A. Stadler, I. Digel, G. Artmann, J. Embs, G. Zaccai, and G. Buldt, "Hemoglobin dynamics in red blood cells: Correlation to body temperature," *Biophysical Journal*, vol. 95, pp. 5449–5461, Dec. 2008.

[4] E. Yus-Najera, I. Santana-Castro, and A. Villarroel, "The identification and characterization of a noncontinuous calmodulin-binding site in noninactivating voltage-dependent KCNQ potassium channels," *The Journal of biological chemistry*, vol. 277, pp. 28545–28553, Aug. 2002. PMID: 12032157.

[5] K. Morrison and G. Weiss, "Combinatorial alanine-scanning," *Current Opinion in Chemical Biology*, vol. 5, no. 3, pp. 302–307, 2001.

[6] Raymond C. Stevens, "The cost and value of three-dimensional protein structure," in *Drug Discovery World*, pp. 35–48, 2003.

[7] H. Ledford, "Consortium solves its 1,000th protein structure," *Nature News*, Sept. 2010.

[8] L. Slabinski, L. Jaroszewski, A. P. C. Rodrigues, L. Rychlewski, I. A. Wilson, S. A. Lesley, and A. Godzik, "The challenge of protein structure determination–lessons from structural genomics," *Protein science: a publication of the Protein Society*, vol. 16, pp. 2472–2482, Nov. 2007. PMID: 17962404.

[9] C. B. Anfinsen, "Principles that govern the folding of protein chains," *Science*, vol. 181, pp. 223–230, July 1973. PMID: 4124164.

[10] R. Chattopadhyaya, W. E. Meador, A. R. Means, and F. A. Quiocho, "Calmodulin structure refined at 1.7 angstroms resolution," *Journal of molecular biology*, vol. 228, pp. 1177–1192, Dec. 1992. PMID: 1474585.

[11] F. C. Bernstein, T. F. Koetzle, G. J. Williams, J. Meyer, E F, M. D. Brice, J. R. Rodgers, O. Kennard, T. Shimanouchi, and M. Tasumi, "The protein data bank: a computer-based archival file for macromolecular structures," *Journal of Molecular Biology*, vol. 112, pp. 535–542, May 1977. PMID: 875032.

[12] G. A. Petsko and D. Ringe, *Protein structure and function*. New Science Press, 2004.

[13] D. C. Ramsey, M. P. Scherrer, T. Zhou, and C. O. Wilke, "The relationship between relative solvent accessibility and evolutionary rate in protein evolution," *Genetics*, vol. 188, pp. 479–488, June 2011. PMID: 21467571 PMCID: PMC3122320.

[14] B. Ma, T. Elkayam, H. Wolfson, and R. Nussinov, "Proteinprotein interactions: Structurally conserved residues distinguish between binding sites and exposed protein surfaces," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 100, pp. 5772–5777, May 2003. PMID: 12730379 PMCID: PMC156276.

[15] S. Mika and B. Rost, "Protein protein interactions more conserved within species than across species," *PLoS Comput Biol*, vol. 2, p. e79, July 2006.

[16] E. Fischer, "Einfluss der configuration auf die wirkung der enzyme," *Berichte der deutschen chemischen Gesellschaft*, vol. 27, pp. 2985–2993, Oct. 1894.

[17] A. Kahraman, R. J. Morris, R. A. Laskowski, and J. M. Thornton, "Shape variation in protein binding pockets and their ligands," *Journal of Molecular Biology*, vol. 368,

pp. 283–301, Apr. 2007.

[18] D. E. Koshland, "Application of a theory of enzyme specificity to protein synthesis," *Proceedings of the National Academy of Sciences*, vol. 44, pp. 98–104, Feb. 1958.

[19] H. R. Bosshard, "Molecular recognition by induced fit: How fit is the concept?," *Physiology*, vol. 16, pp. 171–173, Aug. 2001.

[20] Y. Ofran, "Prediction of protein interaction sites," in *Computational Protein-Protein Interaction*, pp. 167–184, CRC Press, 2009.

[21] Y. Ofran and B. Rost, "Analysing six types of protein-protein interfaces," *Journal of Molecular Biology*, vol. 325, pp. 377–387, Jan. 2003.

[22] A. G. Ngounou Wetie, I. Sokolowska, A. G. Woods, U. Roy, J. A. Loo, and C. C. Darie, "Investigation of stable and transient protein-protein interactions: Past, present, and future," *PROTEOMICS*, vol. 13, no. 3-4, p. 538557, 2013.

[23] Samatha Kottha and Michael Schroeder, "Classifying permanent and transient protein interactions," in *Proceedings of German Bioinformatics Conference*, (Germany), 2006.

[24] S. Jones and J. M. Thornton, "Prediction of protein-protein interaction sites using patch analysis," *Journal of Molecular Biology*, vol. 272, pp. 133–143, Sept. 1997. PMID: 9299343.

[25] J. Liang, H. Edelsbrunner, and C. Woodward, "Anatomy of protein pockets and cavities: measurement of binding site geometry and implications for ligand design," *Protein Science: A Publication of the Protein Society*, vol. 7, pp. 1884–1897, Sept. 1998. PMID: 9761470.

[26] A. Passerini, M. Lippi, and P. Frasconi, "MetalDetector v2.0: predicting the geometry of metal binding sites from protein sequence," *Nucleic Acids Research*, vol. 39,

pp. W288–W292, May 2011.

[27] H. Neuvirth, R. Raz, and G. Schreiber, "ProMate: a structure based prediction program to identify the location of protein-protein binding sites," *Journal of Molecular Biology*, vol. 338, pp. 181–199, Apr. 2004. PMID: 15050833.

[28] J.-L. Chung, W. Wang, and P. E. Bourne, "High-throughput identification of interacting protein-protein binding sites," *BMC Bioinformatics*, vol. 8, p. 223, June 2007.

[29] J.-L. Chung, W. Wang, and P. E. Bourne, "Exploiting sequence and structure homologs to identify protein-protein binding sites," *Proteins*, vol. 62, pp. 630–640, Mar. 2006. PMID: 16329107.

[30] J.-F. Xia, X.-M. Zhao, J. Song, and D.-S. Huang, "APIS: accurate prediction of hot spots in protein interfaces by combining protrusion index with solvent accessibility," *BMC Bioinformatics*, vol. 11, p. 174, Apr. 2010. PMID: 20377884 PMCID: 2874803.

[31] W. Kabsch and C. Sander, "Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features," *Biopolymers*, vol. 22, pp. 2577–2637, Dec. 1983. PMID: 6667333.

[32] S. Liang, C. Zhang, S. Liu, and Y. Zhou, "Protein binding site prediction using an empirical scoring function," *Nucleic Acids Research*, vol. 34, pp. 3698–3707, Jan. 2006.

[33] S. Vajda and F. Guarnieri, "Characterization of protein-ligand interaction sites using experimental and computational methods," *Current Opinion in Drug Discovery and Development*, vol. 9, no. 3, pp. 363–369, 2006.

[34] D. R. Caffrey, S. Somaroo, J. D. Hughes, J. Mintseris, and E. S. Huang, "Are protein-protein interfaces more conserved in sequence than the rest of the protein surface?," *Protein Science: A Publication of the Protein Society*, vol. 13, pp. 190–202, Jan. 2004.

PMID: 14691234.

[35] N. V. Grishin and M. A. Phillips, "The subunit interfaces of oligomeric enzymes are conserved to a similar extent to the overall protein sequences," *Protein Science*, vol. 3, pp. 2455–2458, Dec. 1994.

[36] W. S. J. Valdar, "Scoring residue conservation," *Proteins: Structure, Function, and Bioinformatics*, vol. 48, pp. 227–241, Aug. 2002.

[37] D. Rajamani, S. Thiel, S. Vajda, and C. J. Camacho, "Anchor residues in protein-protein interactions," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 101, pp. 11287–11292, Aug. 2004. PMID: 15269345.

[38] F. Frigerio, A. Coda, L. Pugliese, C. Lionetti, E. Menegatti, G. Amiconi, H. P. Schnebli, P. Ascenzi, and M. Bolognesi, "Crystal and molecular structure of the bovine alpha-chymotrypsin-eglin c complex at 2.0 a resolution," *Journal of molecular biology*, vol. 225, pp. 107–123, May 1992. PMID: 1583684.

[39] L. M. C. Meireles, A. S. Dmling, and C. J. Camacho, "ANCHOR: a web server and database for analysis of protein-protein interaction binding pockets for drug discovery," *Nucleic Acids Research*, vol. 38, pp. W407–W411, July 2010. PMID: 20525787.

[40] Y. Levy and J. N. Onuchic, "Water and proteins: A love-hate relationship," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 101, pp. 3325–3326, Mar. 2004. PMID: 14993602.

[41] S. Qin and H.-X. Zhou, "Meta-PPISP: a meta web server for protein-protein interaction site prediction," *Bioinformatics*, vol. 23, pp. 3386–3387, Dec. 2007.

[42] S. Ahmad and K. Mizuguchi, "Partner-aware prediction of interacting residues in protein-protein complexes from sequence data," *PLoS ONE*, vol. 6, Dec. 2011. PMID:

22194998 PMCID: 3237601.

[43] R. Chen, L. Li, and Z. Weng, "ZDOCK: an initial-stage protein-docking algorithm," *Proteins: Structure, Function, and Bioinformatics*, vol. 52, no. 1, p. 8087, 2003.

[44] I. Ezkurdia, L. Bartoli, P. Fariselli, R. Casadio, A. Valencia, and M. L. Tress, "Progress and challenges in predicting protein-protein interaction sites," *Briefings in Bioinformatics*, vol. 10, pp. 233–246, May 2009.

[45] M. N. Wass, A. David, and M. J. Sternberg, "Challenges for the prediction of macromolecular interactions," *Current Opinion in Structural Biology*, vol. 21, pp. 382–390, June 2011.

[46] S. J. de Vries and A. M. J. J. Bonvin, "How proteins get in touch: interface prediction in the study of biomolecular complexes," *Current Protein & Peptide Science*, vol. 9, pp. 394–406, Aug. 2008. PMID: 18691126.

[47] H. Hwang, T. Vreven, J. Janin, and Z. Weng, "Protein-protein docking benchmark version 4.0," *Proteins: Structure, Function, and Bioinformatics*, vol. 78, no. 15, p. 31113114, 2010.

[48] N. Tuncbag, G. Kar, A. Gursoy, O. Keskin, and R. Nussinov, "Towards inferring time dimensionality in protein-protein interaction networks by integrating structures: the p53 example," *Molecular bioSystems*, vol. 5, pp. 1770–1778, Dec. 2009. PMID: 19585003.

[49] W. K. Kim, A. Henschel, C. Winter, and M. Schroeder, "The many faces of protein-protein interactions: A compendium of interface geometry," *PLoS Comput Biol*, vol. 2, p. e124, Sept. 2006.

[50] P. J. Kundrotas and I. A. Vakser, "Protein-protein alternative binding modes do not overlap," *Protein Science*, pp. 1141–5, 2013.

[51] J. Marsh and S. Teichmann, "Relative solvent accessible surface area predicts protein conformational changes upon binding," *Structure*, vol. 19, pp. 859–867, June 2011.

[52] J. Bray, D. Weiss, and M. Levitt, "Optimized torsion-angle normal modes reproduce conformational changes more accurately than cartesian modes," *Biophysical Journal*, vol. 101, pp. 2966–2969, Dec. 2011.

[53] G. R. Smith, M. J. Sternberg, and P. A. Bates, "The relationship between the flexibility of proteins and their conformational states on forming protein-protein complexes with an application to protein-protein docking," *Journal of Molecular Biology*, vol. 347, pp. 1077–1101, Apr. 2005.

[54] S. Rackovsky, "Global characteristics of protein sequences and their implications," *Proceedings of the National Academy of Sciences*, vol. 107, pp. 8623–8626, May 2010.

[55] R. P. Bahadur, P. Chakrabarti, F. Rodier, and J. Janin, "A dissection of specific and non-specific protein-protein interfaces," *Journal of molecular biology*, vol. 336, pp. 943–955, Feb. 2004. PMID: 15095871.

[56] J. Martin, "Beauty is in the eye of the beholder: proteins can recognize binding sites of homologous proteins in more than one way," *PLoS computational biology*, vol. 6, no. 6, p. e1000821, 2010.

[57] H. Nishi, K. Hashimoto, and A. Panchenko, "Phosphorylation in protein-protein binding: Effect on stability and function," *Structure*, vol. 19, pp. 1807–1815, Dec. 2011.

[58] V. Vagenende, A. X. Han, H. B. Pek, and B. L. W. Loo, "Quantifying the molecular origins of opposite solvent effects on protein-protein interactions," *PLoS Comput Biol*,

vol. 9, p. e1003072, May 2013.

[59] J. Siltberg-Liberles, J. A. Grahnen, and D. A. Liberles, "The evolution of protein structures and structural ensembles under functional constraint," *Genes*, vol. 2, pp. 748–762, Oct. 2011.

[60] M. M. Gromiha, N. Saranya, S. Selvaraj, B. Jayaram, and K. Fukui, "Sequence and structural features of binding site residues in protein-protein complexes: comparison with protein-nucleic acid complexes," *Proteome Science*, vol. 9, p. S13, Oct. 2011.

[61] J. Janin, "Specific versus non-specific contacts in protein crystals," *Nature Structural & Molecular Biology*, vol. 4, pp. 973–974, Dec. 1997.

[62] L. Garma, S. Mukherjee, P. Mitra, and Y. Zhang, "How many protein-protein interactions types exist in nature?," *PLoS ONE*, vol. 7, p. e38913, June 2012.

[63] S. Jones and J. M. Thornton, "Analysis of protein-protein interaction sites using surface patches," *Journal of molecular biology*, vol. 272, pp. 121–132, Sept. 1997. PMID: 9299342.

[64] Q. C. Zhang, D. Petrey, R. Norel, and B. H. Honig, "Protein interface conservation across structure space," *Proceedings of the National Academy of Sciences*, June 2010.

[65] H.-X. Zhou and S. Qin, "Interaction-site prediction for protein complexes: a critical assessment," *Bioinformatics (Oxford, England)*, vol. 23, pp. 2203–2209, Sept. 2007. PMID: 17586545.

[66] S. Leis, S. Schneider, and M. Zacharias, "In silico prediction of binding sites on proteins," *Current Medicinal Chemistry*, vol. 2010, no. 17, pp. 1550–1562, 2010.

[67] Q. C. Zhang, L. Deng, M. Fisher, J. Guan, B. Honig, and D. Petrey, "PredUs: a web server for predicting protein interfaces using structural neighbors," *Nucleic Acids*

Research, vol. 39, pp. W283–W287, May 2011.

[68] R. A. Jordan, Y. EL-Manzalawy, D. Dobbs, and V. Honavar, "Predicting protein-protein interface residues using local surface structural similarity," *BMC Bioinformatics*, vol. 13, no. 1, p. 41, 2012.

[69] H. Chen and H. Zhou, "Prediction of interface residues in proteinprotein complexes by a consensus neural network method: Test against NMR data," *Proteins: Structure, Function, and Bioinformatics*, vol. 61, pp. 21–35, Oct. 2005.

[70] S. Vajda, D. R. Hall, and D. Kozakov, "Sampling and scoring: A marriage made in heaven," *Proteins: Structure, Function, and Bioinformatics*, p. n/an/a, 2013.

[71] B. G. Pierce, Y. Hourai, and Z. Weng, "Accelerating protein docking in ZDOCK using an advanced 3D convolution library," *PLoS ONE*, vol. 6, p. e24657, Sept. 2011.

[72] C. Bajaj, R. Chowdhury, and V. Siddahanavalli, "F2Dock: fast fourier protein-protein docking," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 8, pp. 45 –58, Feb. 2011.

[73] O. Schueler-Furman, C. Wang, and D. Baker, "Progress in proteinprotein docking: Atomic resolution predictions in the CAPRI experiment using RosettaDock with an improved treatment of side-chain flexibility," *Proteins: Structure, Function, and Bioinformatics*, vol. 60, no. 2, p. 187194, 2005.

[74] S. J. de Vries, A. D. J. van Dijk, M. Krzeminski, M. van Dijk, A. Thureau, V. Hsu, T. Wassenaar, and A. M. J. J. Bonvin, "HADDOCK versus HADDOCK: new features and performance of HADDOCK2.0 on the CAPRI targets," *Proteins*, vol. 69, pp. 726–733, Dec. 2007. PMID: 17803234.

[75] B. Huang and M. Schroeder, "Using protein binding site prediction to improve protein docking," *Gene*, vol. 422, pp. 14–21, Oct. 2008.

[76] J. Mintseris, B. Pierce, K. Wiehe, R. Anderson, R. Chen, and Z. Weng, "Integrating statistical pair potentials into protein complex prediction," *Proteins: Structure, Function, and Bioinformatics*, vol. 69, no. 3, p. 511520, 2007.

[77] A. Amos-Binks, C. Patulea, S. Pitre, A. Schoenrock, Y. Gui, J. R. Green, A. Golshani, and F. Dehne, "Binding site prediction for protein-protein interactions and novel motif discovery using re-occurring polypeptide sequences," *BMC Bioinformatics*, vol. 12, p. 225, June 2011.

[78] R. Sinha, P. J. Kundrotas, and I. Vakser, "Docking by structural similarity at protein-protein interfaces," *Proteins*, vol. 78, pp. 3235–3241, Nov. 2010. PMID: 20715056 PMCID: PMC2952659.

[79] O. Keskin, R. Nussinov, and A. Gursoy, "PRISM: protein-protein interaction prediction by structural matching," *Methods in molecular biology (Clifton, N.J.)*, vol. 484, pp. 505–521, 2008. PMID: 18592198.

[80] S. Gnther, P. May, A. Hoppe, C. Frmmel, and R. Preissner, "Docking without docking: ISEARCHprediction of interactions using known interfaces," *Proteins: Structure, Function, and Bioinformatics*, vol. 69, no. 4, p. 839844, 2007.

[81] P. J. Kundrotas, Z. Zhu, J. Janin, and I. A. Vakser, "Templates are available to model nearly all complexes of structurally characterized proteins," *Proceedings of the National Academy of Sciences*, May 2012. PMID: 22645367.

[82] Y. Zhang and J. Skolnick, "TM-align: a protein structure alignment algorithm based on the TM-score," *Nucleic Acids Research*, vol. 33, pp. 2302–2309, Jan. 2005. PMID:

15849316.

[83] P. J. Kundrotas, M. F. Lensink, and E. Alexov, "Homology-based modeling of 3D structures of proteinprotein complexes using alignments of modified sequence profiles," *International Journal of Biological Macromolecules*, vol. 43, pp. 198–208, Aug. 2008.

[84] T. Vreven, H. Hwang, B. G. Pierce, and Z. Weng, "Evaluating template-based and template-free protein-protein complex structure prediction," *Briefings in Bioinformatics*, July 2013. PMID: 23818491.

[85] H. Wang, E. Segal, A. Ben-Hur, Q.-R. Li, M. Vidal, and D. Koller, "InSite: a computational method for identifying protein-protein interaction binding sites on a proteome-wide scale," *Genome Biology*, vol. 8, no. 9, p. R192, 2007. PMID: 17868464 PMCID: 2375030.

[86] F. u. A. Afsar Minhas, B. J. Geiss, and A. Ben-Hur, "PAIRpred: partner-specific prediction of interacting residues from sequence and structure," *Proteins: Structure, Function, and Bioinformatics*, p. n/an/a, 2013.

[87] J. Janin, "Docking predictions of protein-protein interactions and their assessment: The CAPRI experiment," in *Identification of Ligand Binding Site and Protein-Protein Interaction Area* (I. Roterman-Konieczna, ed.), no. 8 in Focus on Structural Biology, pp. 87–104, Springer Netherlands, Jan. 2013.

[88] A. Andreeva, D. Howorth, S. E. Brenner, T. J. P. Hubbard, C. Chothia, and A. G. Murzin, "SCOP database in 2004: refinements integrate structure and sequence family data," *Nucleic acids research*, vol. 32, pp. D226–229, Jan. 2004. PMID: 14681400.

[89] H. Hwang, B. Pierce, J. Mintseris, J. Janin, and Z. Weng, "Protein-protein docking benchmark version 3.0," *Proteins: Structure, Function, and Bioinformatics*, vol. 73,

no. 3, p. 705709, 2008.

[90] D. Frishman and P. Argos, "Knowledge-based protein secondary structure assignment," *Proteins*, vol. 23, p. 566579, 1995.

[91] M. Sanner, A. Olson, and J.-C. Spehner, "Reduced surface: an efficient way to compute molecular surfaces," *Biopolymers*, vol. 38, no. 3, pp. 305–320, 1996.

[92] T. Hamelryck, "An amino acid has two sides: A new 2D measure provides a different view of solvent exposure," *Proteins: Structure, Function, and Bioinformatics*, vol. 59, no. 1, p. 3848, 2005.

[93] A. Pintar, O. Carugo, and S. Pongor, "CX, an algorithm that identifies protruding atoms in proteins," *Bioinformatics (Oxford, England)*, vol. 18, pp. 980–984, July 2002. PMID: 12117796.

[94] J. Mihel, M. iki, S. Tomi, B. Jeren, and K. Vlahoviek, "PSAIA protein structure and interaction analyzer," *BMC Structural Biology*, vol. 8, p. 21, Apr. 2008.

[95] S. F. Altschul, T. L. Madden, A. A. Schffer, J. Zhang, Z. Zhang, W. Miller, and D. J. Lipman, "Gapped BLAST and PSI-BLAST: a new generation of protein database search programs.," *Nucleic Acids Research*, vol. 25, pp. 3389–3402, Sept. 1997. PMID: 9254694 PMCID: 146917.

[96] S. McGinnis and T. L. Madden, "BLAST: at the core of a powerful and diverse set of sequence analysis tools," *Nucleic Acids Research*, vol. 32, pp. W20–W25, July 2004.

[97] M. Remmert, A. Biegert, A. Hauser, and J. Sding, "HHblits: lightning-fast iterative protein sequence searching by HMM-HMM alignment," *Nature Methods*, vol. 9, pp. 173–175, Feb. 2012.

[98] E. Faraggi, T. Zhang, Y. Yang, L. Kurgan, and Y. Zhou, "SPINE x: Improving protein secondary structure prediction by multistep learning coupled with prediction of solvent accessible surface area and backbone torsion angles," *Journal of Computational Chemistry*, vol. 33, pp. 259–267, Jan. 2012.

[99] C. Cortes and V. Vapnik, "Support-vector networks," in *Machine Learning*, p. 273297, 1995.

[100] A. Ben-Hur, C. S. Ong, S. Sonnenburg, B. Schlkopf, and G. Rtsch, "Support vector machines and kernels for computational biology," *PLoS Comput Biol*, vol. 4, p. e1000173, Oct. 2008.

[101] A. Ben-Hur and W. S. Noble, "Kernel methods for predicting protein-protein interactions," *Bioinformatics*, vol. 21, pp. i38–i46, June 2005.

[102] J.-P. Vert, J. Qiu, and W. Noble, "A new pairwise kernel for biological network inference with support vector machines," *BMC Bioinformatics*, vol. 8, p. S8, Dec. 2007.

[103] A. Bar-hillel and D. Weinshall, "Boosting margin based distance functions for clustering," in *In Proceedings of the Twenty-First International Conference on Machine Learning*, p. 393400, 2004.

[104] C. Brunner, A. Fischer, K. Luig, and T. Thies, "Pairwise support vector machines and their application to large scale problems," *Journal of Machine Learning Research*, vol. 13, p. 22792292, Aug. 2012.

[105] C. Brunner, A. Fischer, K. Luig, and T. Thies, "Pairwise kernels, support vector machines, and the application to large scale problems," tech. rep., Technische Universitat Dresden Institut fur Numerische Mathematik, 2011.

[106] S. S. . Haykin, *Neural networks a comprehensive foundation*. Prentice Hall, 2nd ed. ed., 1999.

[107] A. Ben-Hur, "PyML: machine learning using python (http://pyml.sourceforge.net/)," Dec. 2012.

[108] M. F. Lensink and S. J. Wodak, "Docking and scoring protein interactions: CAPRI 2009," *Proteins: Structure, Function, and Bioinformatics*, vol. 78, no. 15, p. 30733084, 2010.

[109] R. Guan, L.-C. Ma, P. G. Leonard, B. R. Amer, H. Sridharan, C. Zhao, R. M. Krug, and G. T. Montelione, "Structural basis for the sequence-specific recognition of human ISG15 by the NS1 protein of influenza b virus," *Proceedings of the National Academy of Sciences*, vol. 108, pp. 13468–13473, Aug. 2011.

[110] C. Yin, J. A. Khan, G. V. T. Swapna, A. Ertekin, R. M. Krug, L. Tong, and G. T. Montelione, "Conserved surface features form the double-stranded RNA binding site of non-structural protein 1 (NS1) from influenza a and b viruses," *Journal of Biological Chemistry*, vol. 282, pp. 20584–20592, May 2007.

[111] J. Narasimhan, "Crystal structure of the interferon-induced ubiquitin-like protein ISG15," *Journal of Biological Chemistry*, vol. 280, pp. 27356–27365, June 2005.

[112] R. Guan, L.-C. Ma, P. G. Leonard, B. R. Amer, H. Sridharan, C. Zhao, R. M. Krug, and G. T. Montelione, "Structural basis for the sequence-specific recognition of human ISG15 by the NS1 protein of influenza b virus," *Proceedings of the National Academy of Sciences*, vol. 108, pp. 13468–13473, Aug. 2011.

[113] Wikipedia contributors, "Structural alignment software (http://en.wikipedia.org)," May 2013. Page Version ID: 556339759.

[114] M. Shatsky, R. Nussinov, and H. J. Wolfson, "MultiProt a multiple protein structural alignment algorithm," in *Algorithms in Bioinformatics* (R. Guig and D. Gusfield, eds.), vol. 2452, pp. 235–250, Berlin, Heidelberg: Springer Berlin Heidelberg.

[115] J. Konc and D. Janezic, "ProBiS algorithm for detection of structurally similar protein binding sites by local structural alignment," *Bioinformatics*, vol. 26, pp. 1160–1168, May 2010.

[116] J. Konc and D. Janezic, "ProBiS-2012: web server and web services for detection of structurally similar binding sites in proteins," *Nucleic acids research*, vol. 40, pp. W214–221, July 2012. PMID: 22600737.

[117] J. Konc and D. Janei, "ProteinProtein binding-sites prediction by protein surface structure conservation," *Journal of Chemical Information and Modeling*, vol. 47, pp. 940–944, May 2007.

[118] N. Tuncbag, A. Gursoy, R. Nussinov, and O. Keskin, "Predicting protein-protein interactions on a proteome scale by matching evolutionary and structural similarities at interfaces using PRISM," *Nature Protocols*, vol. 6, pp. 1341–1354, Sept. 2011.

[119] T. F. Smith and M. S. Waterman, "Identification of common molecular subsequences," *Journal of molecular biology*, vol. 147, pp. 195–197, Mar. 1981. PMID: 7265238.

[120] Dmitry Pechyony, *Theory and Practice of Transductive Learning*. PhD thesis, Technion, Israel, 2008.

[121] Chapelle, Olivier, Sch{\"o}lkopf, Bernhard, and Zien, Alexander, *Semi-supervised learning*. MIT press Cambridge, 2006.

[122] G. Ifrim and G. Weikum, "Transductive learning for text classification using explicit knowledge models," in *In PKDD*, 2006.

[123] R. El-Yaniv, D. Pechyony, and V. Vapnik, "Large margin vs. large volume in transductive learning," *Mach. Learn.*, vol. 72, p. 173188, Sept. 2008.

[124] A. Levin, D. Lischinski, and Y. Weiss, "Colorization using optimization," *ACM Trans. Graph.*, vol. 23, p. 689694, Aug. 2004.

[125] A. Blake, C. Rother, M. Brown, P. Perez, and P. Torr, "Interactive image segmentation using an adaptive GMMRF model," *IN ECCV*, vol. 1, pp. 428—441, 2004.

[126] K. Duh and K. Kirchhoff, "Lexicon acquisition for dialectal arabic using transductive learning," in *In Proceedings of EMNLP*, 2006.

[127] N. Ueffing, "Transductive learning for statistical machine translation," in *In Proc. of ACL*, p. 2532, 2007.

[128] N. Kasabov and S. Pang, "Transductive support vector machines and applications in bioinformatics for promoter recognition," in *Proceedings of the 2003 International Conference on Neural Networks and Signal Processing, 2003*, vol. 1, pp. 1–6 Vol.1, 2003.

[129] S. Lise, C. Archambeau, M. Pontil, and D. T. Jones, "Prediction of hot spot residues at protein-protein interfaces by combining machine learning and energy-based methods," *BMC Bioinformatics*, vol. 10, p. 365, Oct. 2009. PMID: 19878545.

[130] R. A. Craig and L. Liao, "Transductive learning with EM algorithm to classify proteins based on phylogenetic profiles," *Int. J. Data Min. Bioinformatics*, vol. 1, p. 337351, Apr. 2007.

[131] A. Sokolov and A. Ben-Hur, "Multi-view prediction of protein function," in *Proceedings of the 2nd ACM Conference on Bioinformatics, Computational Biology and Biomedicine*, BCB '11, (New York, NY, USA), p. 135142, ACM, 2011.

[132] G. Yu, C. Domeniconi, H. Rangwala, G. Zhang, and Z. Yu, "Transductive multi-label ensemble classification for protein function prediction," in *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '12, (New York, NY, USA), p. 10771085, ACM, 2012.

[133] G. Erkan, "Semi-supervised classification for extracting protein interaction sentences using dependency parsing," in *In Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL*, p. 228237, 2007.

[134] T. Joachims, "Transductive inference for text classification using support vector machines," p. 200209, Morgan Kaufmann, 1999.

[135] O. Chapelle, V. Sindhwani, and S. S. Keerthi, "Optimization techniques for semi-supervised support vector machines," *J. Mach. Learn. Res.*, vol. 9, p. 203233, June 2008.

[136] V. Sindhwani and S. S. Keerthi, "Large scale semi-supervised linear SVMs," in *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '06, (New York, NY, USA), p. 477484, ACM, 2006.

[137] R. Yan, J. Zhang, J. Yang, and A. G. Hauptmann, "A discriminative learning framework with pairwise constraints for video object classification," *IEEE transactions on pattern analysis and machine intelligence*, vol. 28, pp. 578–593, Apr. 2006. PMID: 16566507.

[138] H. Zeng and Y.-M. Cheung, "Semi-supervised maximum margin clustering with pairwise constraints," *IEEE Transactions on Knowledge and Data Engineering*, vol. 24,

no. 5, pp. 926–939, 2012.

[139] J. Davis, B. Kulis, S. Sra, and I. Dhillon, "Information-theoretic metric learning," in *in NIPS 2006 Workshop on Learning to Compare Examples*, 2007.

[140] A. Globerson and S. Roweis, *Metric Learning by Collapsing Classes*.

[141] J. Goldberger, S. Roweis, G. Hinton, and R. Salakhutdinov, "Neighbourhood components analysis," in *Advances in Neural Information Processing Systems 17*, p. 513520, MIT Press, 2004.

[142] S. Shalev-Shwartz, Y. Singer, and A. Y. Ng, "Online and batch learning of pseudo-metrics," in *Proceedings of the twenty-first international conference on Machine learning*, ICML '04, (New York, NY, USA), p. 94, ACM, 2004.

[143] K. Q. Weinberger, J. Blitzer, and L. K. Saul, "Distance metric learning for large margin nearest neighbor classification," in *In NIPS*, MIT Press, 2006.

[144] J. Zhang and R. Yan, *On the Value of Pairwise Constraints in Classification and Consistency*.

[145] N. Nguyen and R. Caruana, "Improving classification with pairwise constraints: A margin-based approach," in *Machine Learning and Knowledge Discovery in Databases* (W. Daelemans, B. Goethals, and K. Morik, eds.), no. 5212 in Lecture Notes in Computer Science, pp. 113–124, Springer Berlin Heidelberg, Jan. 2008.

[146] O. Chapelle and A. Zien, "Semi-supervised classification by low density separation," Sept. 2004.

[147] B. Schlkopf and A. J. Smola, *Learning With Kernels: Support Vector Machines, Regularization, Optimization and Beyond*. MIT Press, 2002.

[148] S. Shalev-Shwartz, Y. Singer, N. Srebro, and A. Cotter, "Pegasos: primal estimated sub-gradient solver for SVM," *Mathematical Programming*, vol. 127, pp. 3–30, Mar. 2011.

[149] J. C. Platt, "Sequential minimal optimization: A fast algorithm for training support vector machines," tech. rep., ADVANCES IN KERNEL METHODS - SUPPORT VECTOR LEARNING, 1998.

[150] A. Ben-Hur and W. S. Noble, "Kernel methods for predicting protein-protein interactions," *Bioinformatics*, vol. 21, p. 3846, Jan. 2005.

[151] N. Bouch, A. Yellin, W. A. Snedden, and H. Fromm, "Plant-specific calmodulin-binding proteins," *Annual Review of Plant Biology*, vol. 56, no. 1, pp. 435–466, 2005. PMID: 15862103.

[152] A. Reddy, A. Ben-Hur, and I. S. Day, "Experimental and computational approaches for the study of calmodulin interactions," *Phytochemistry*, vol. 72, pp. 1007–1019, July 2011.

[153] F. u. A. A. Minhas and A. Ben-Hur, "Multiple instance learning of calmodulin binding sites," *Bioinformatics*, vol. 28, pp. i416–i422, Sept. 2012.

[154] K. L. Yap, J. Kim, K. Truong, M. Sherman, T. Yuan, and M. Ikura, "Calmodulin target database," *Journal of Structural and Functional Genomics*, vol. 1, pp. 8–14, Mar. 2000.

[155] K. T. O'Neil and W. F. DeGrado, "How calmodulin binds its targets: sequence independent recognition of amphiphilic -helices," *Trends in Biochemical Sciences*, vol. 15, pp. 59–64, Feb. 1990.

[156] M. Hamilton, A. Reddy, and A. Ben-Hur, "Kernel methods for calmodulin binding and binding site prediction," in *Proceedings of the 2nd ACM Conference on Bioinformatics, Computational Biology and Biomedicine*, p. 381386, 2011.

[157] P. Radivojac, S. Vucetic, T. R. Oconnor, V. N. Uversky, Z. Obradovic, and A. K. Dunker, "Calmodulin signaling: analysis and prediction of a disorder-dependent molecular recognition," *Proteins*, vol. 63, p. 398410, 2006.

[158] T. G. Dietterich, R. H. Lathrop, T. Lozano-Perez, and A. Pharmaceutical, "Solving the multiple-instance problem with axis-parallel rectangles," *Artificial Intelligence*, vol. 89, p. 3171, 1997.

[159] B. Babenko, M.-H. Yang, and S. Belongie, "Robust object tracking with online multiple instance learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, p. 16191632, Aug. 2011.

[160] Q. Tao, S. Scott, N. V. Vinodchandran, and T. T. Osugi, "SVM-based generalized multiple-instance learning via approximate box counting," in *In Proceedings of the Twenty-First International Conference on Machine Learning*, p. 779806, Morgan Kaufmann, 2004.

[161] R. Teramoto and H. Kashima, "Prediction of proteinligand binding affinities using multiple instance learning," *Journal of Molecular Graphics and Modelling*, vol. 29, pp. 492–497, Nov. 2010.

[162] S. C. Popescu, G. V. Popescu, S. Bachan, Z. Zhang, M. Seay, M. Gerstein, M. Snyder, and S. P. Dinesh-Kumar, "Differential binding of calmodulin-related proteins to their targets revealed through high-density arabidopsis protein microarrays," *Proceedings of the National Academy of Sciences*, vol. 104, pp. 4730–4735, Mar. 2007. PMID:

17360592.

[163] S. Andrews, I. Tsochantaridis, and T. Hofmann, "Support vector machines for multiple-instance learning," in *Advances in Neural Information Processing Systems 15*, p. 561568, MIT Press, 2003.

[164] T. Joachims, "Training linear SVMs in linear time," in *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '06, (New York, NY, USA), p. 217226, ACM, 2006.

[165] C. Leslie, E. Eskin, and W. S. Noble, "The spectrum kernel: a string kernel for SVM protein classification," pp. 566–575, 2002.

[166] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman, "Basic local alignment search tool," *Journal of molecular biology*, vol. 215, pp. 403–410, Oct. 1990. PMID: 2231712.

[167] E. Eden, R. Navon, I. Steinfeld, D. Lipson, and Z. Yakhini, "GOrilla: a tool for discovery and visualization of enriched GO terms in ranked gene lists," *BMC Bioinformatics*, vol. 10, p. 48, Feb. 2009. PMID: 19192299.

[168] L. C. Xue, R. A. Jordan, Y. El-Manzalawy, D. Dobbs, and V. Honavar, "Ranking docked models of protein-protein complexes using predicted partner-specific protein-protein interfaces: a preliminary study," in *Proceedings of the 2nd ACM Conference on Bioinformatics, Computational Biology and Biomedicine*, BCB '11, (New York, NY, USA), p. 441445, ACM, 2011.

[169] D. Kozakov, R. Brenke, S. R. Comeau, and S. Vajda, "PIPER: an FFT-based protein docking program with pairwise potentials," *Proteins: Structure, Function, and Bioinformatics*, vol. 65, no. 2, p. 392406, 2006.

[170] C. Winter, A. Henschel, W. K. Kim, and M. Schroeder, "SCOPPI: a structural classification of proteinprotein interfaces," *Nucleic Acids Research*, vol. 34, pp. D310–D314, Jan. 2006. PMID: 16381874.

[171] E. D. Levy, J. B. Pereira-Leal, C. Chothia, and S. A. Teichmann, "3D complex: A structural classification of protein complexes," *PLoS Comput Biol*, vol. 2, p. e155, Nov. 2006.

[172] C. Snchez Claros and A. Tramontano, "Detecting mutually exclusive interactions in protein-protein interaction maps," *PLoS ONE*, vol. 7, p. e38765, June 2012.

[173] W. K. Kim, A. Henschel, C. Winter, and M. Schroeder, "The many faces of protein-protein interactions: A compendium of interface geometry," *PLoS Comput Biol*, vol. 2, p. e124, Sept. 2006.

[174] P. M. Kim, L. J. Lu, Y. Xia, and M. B. Gerstein, "Relating three-dimensional structures to protein networks provides evolutionary insights," *Science*, vol. 314, pp. 1938–1941, Dec. 2006.

[175] C.-J. Tsai, B. Ma, and R. Nussinov, "Protein-protein interaction networks: how can a hub protein bind so many different partners?," *Trends in biochemical sciences*, vol. 34, pp. 594–600, Dec. 2009. PMID: 19837592.

[176] D. Ekman, S. Light, s. K. Bjrklund, and A. Elofsson, "What properties characterize the hub proteins of the protein-protein interaction network of saccharomyces cerevisiae?," *Genome Biology*, vol. 7, p. R45, June 2006.

[177] M. Iqbal, A. A. Freitas, C. G. Johnson, and M. Vergassola, "Message-passing algorithms for the prediction of protein domain interactions from protein-protein interaction data," *Bioinformatics*, vol. 24, pp. 2064–2070, Sept. 2008. PMID: 18641010.

[178] S. M. Rawlinson, M. J. Pryor, P. J. Wright, and D. A. Jans, "Dengue virus RNA polymerase NS5: a potential therapeutic target?," *Current Drug Targets*, vol. 7, no. 12, pp. 1623–1638, 2006.

[179] M. Bollati, K. Alvarez, R. Assenberg, C. Baronti, B. Canard, S. Cook, B. Coutard, E. Decroly, X. de Lamballerie, E. A. Gould, G. Grard, J. M. Grimes, R. Hilgenfeld, A. M. Jansson, H. Malet, E. J. Mancini, E. Mastrangelo, A. Mattevi, M. Milani, G. Moureau, J. Neyts, R. J. Owens, J. Ren, B. Selisko, S. Speroni, H. Steuber, D. I. Stuart, T. Unge, and M. Bolognesi, "Structure and functionality in flavivirus NS-proteins: perspectives for drug design," *Antiviral Research*, vol. 87, pp. 125–148, Aug. 2010.

[180] M. Johansson, A. J. Brooks, D. A. Jans, and S. G. Vasudevan, "A small region of the dengue virus-encoded RNA-dependent RNA polymerase, NS5, confers interaction with both the nuclear transport receptor importin- and the viral helicase, NS3," *Journal of General Virology*, vol. 82, pp. 735–745, Apr. 2001.

[181] W. J. Liu, P. L. Sedlak, N. Kondratieva, and A. A. Khromykh, "Complementation analysis of the flavivirus kunjin NS3 and NS5 proteins defines the minimal regions essential for formation of a replication complex and shows a requirement of NS3 in cis for virus assembly," *Journal of Virology*, vol. 76, pp. 10766–10775, Nov. 2002. PMID: 12368319.

[182] A. Kidera, Y. Konishi, M. Oka, T. Ooi, and H. A. Scheraga, "Statistical analysis of the physical properties of the 20 naturally occurring amino acids," *Journal of Protein Chemistry*, vol. 4, pp. 23–55, Feb. 1985.

[183] M. Vetterli, J. Kovaevi, V. K. Goyal, C. c. M. Vetterli, J. Kovaevi, and V. K. Goyal, *The World of Fourier and Wavelets: Theory, Algorithms and Applications. http://FourierAndWavelets.org*. 2005.

[184] Huang, N. E., *Hilbert-Huang Transform and Its Applications*. World Scientific, 2005.

[185] N. Rehman and D. P. Mandic, "Multivariate empirical mode decomposition," *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Science*, Dec. 2009.

[186] J. Konc and D. Janezic, "An improved branch and bound algorithm for the maximum clique problem," *MATCH Communications in Mathematical and in Computer Chemistry*, June 2007.

APPENDIX A

ProBiS

Given two proteins A and B, the objective of ProBiS is to find out how similar the local neighborhoods of two surface accessible residues a and b on the two respective proteins are to each other in terms of their local geometry and physiochemical characteristics. For the purpose of this discussion, it is convenient to represent a residue a on A by $a \equiv (\boldsymbol{v}_a, L(a))$, where $\boldsymbol{v}_a$ represents the position vector of a and $L(a)$ is an integer label indicating the general physiochemical characteristics of the residue (1. H-Bond acceptor, 2. H-Bond donor, 3. Mixed Donor - Acceptor, 4. Aliphatic and 5. Aromatic). The local neighborhood of a residue a is denoted by $N(a)$ and is composed of all surface accessible residues within an inter-atomic distance of 12 Å from a. There are four major steps in ProBiS for determining the structural similarity between a and b as shown in Figure A.1:

I. Determine, using a computationally efficient quantitative measure $s(a, b)$, whether the local neighborhoods of the two residues are similar enough in their geometrical arrangement so as to warrant the computationally complex local alignment or not. Formally, this measure is calculated using the following three steps [117]: (1) construct a 5×48 dimensional local neighborhood descriptor matrix $M_{i,j}^{p \in P}$ for each residue $p \in \{a, b\}$ so that $M_{i,j}^{p \in P}$ represents the number of residues $p^{'}$ in $N(p)$ with $L(p^{'}) = i$ such that $j = \lceil 4 \times d(p, p^{'}) \rceil$. Here, $d(p, p^{'})$ is the distance between the two residues p and $p^{'}$. Note that the maximum distance between p and p' is 12 Å. (2) *Smooth* the counts in the matrices by using $M_{i,j}^{p \in P} = 0.25 M_{i,j-1}^{p \in P} + 0.75 M_{i,j}^{p \in P} + 0.25 M_{i,j+1}^{p \in P}$. (3) Compute the local similarity as $s(a, b) = \dfrac{\sum_{i,j} |M_{ij}^{a \in A} + M_{ij}^{b \in B}|}{\sum_{i,j} |M_{ij}^{a \in A} - M_{ij}^{b \in B}|}$ unless

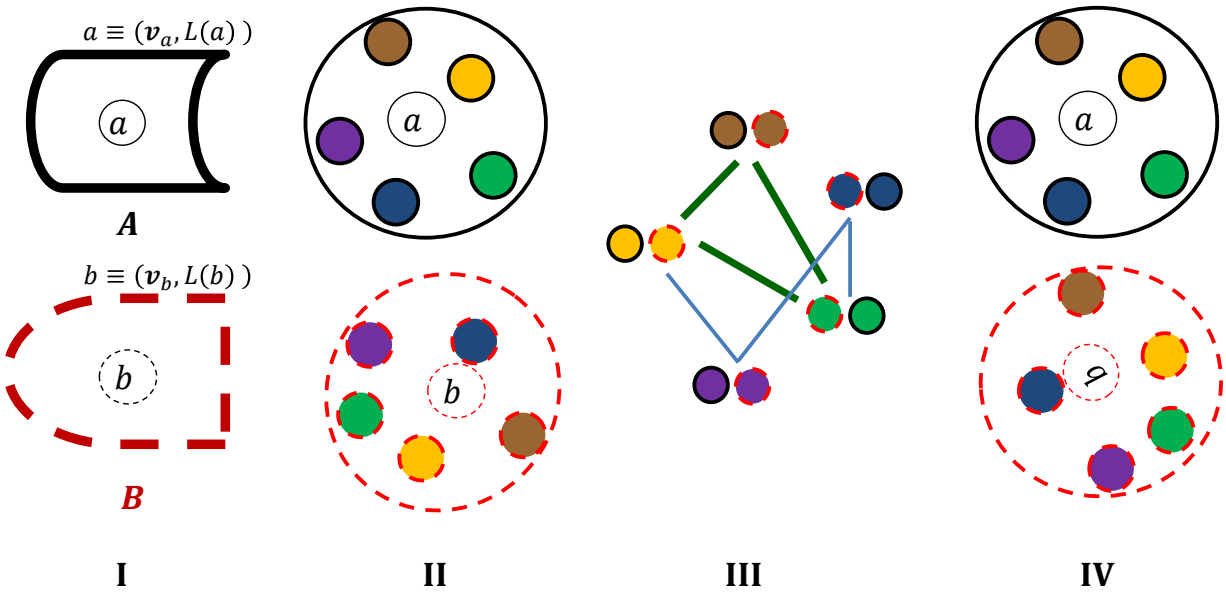


FIGURE A.1. Different steps in computing the local alignment of residues $a \in A$ and $b \in B$ with ProBiS [115]. See the text for details on each step. The local neighborhoods of the two residues are shown in (II). The colors of the residues in (II) indicate their physiochemical label. (III) shows the graph G_{ab}. Note that a vertex in this graph is a pair of identically labeled residues from the two proteins (see equation 20). Two vertices are linked by edges (solid lines) only when the distance between the residues from each protein in the two vertices is almost identical (see equation 21). The vertices in the maximum clique for this graph (of size 3) are linked by edges indicated by thicker lines in green. (IV) shows the transformed neighborhood of b.

the denominator is zero, in which case, this similarity is set equal to the numerator. The following local structural alignment steps (II-IV) are performed only if $s(a,b) > 2.8$.

II. Construct a joint graph $G_{ab} \equiv (V(a,b), E(a,b))$ of the two neighborhoods of residues $a \in A$ and $b \in B$ with vertex and edge sets defined as follows:

$$V(a,b) = \{(a',b') \mid a' \in N(a) \ \wedge \ b' \in N(b) \ \wedge \ L(a') = L(b')\} \tag{20}$$

$$E(a,b) = \{((a',b'),(a'',b'')) \mid |d(a',a'') - d(b',b'')| < 0.5\text{Å} \wedge (a',b'),(a'',b'') \in V(a,b)\} \tag{21}$$

This graph can be viewed as a representation of all possible rotations and translations of the neighborhoods around the residues from the two proteins.

III. Find the maximum clique[13] in G_{ab} using the efficient branch and bound heuristic algorithm given in [186]. A maximal clique in G_{ab} corresponds to an optimal local alignment, i.e., a transformation that superimposes or aligns the largest number of residues with similar physiochemical properties in the two neighborhoods.

IV. Based on the maximum clique, geometrically transform $N(b)$ to match $N(a)$. The alignment is considered significant only if the RMSD after alignment of the two neighborhoods is less than 2.0 Å and there are 10 or more vertices in the alignment.

[13]A clique is a subset of vertices of a graph such that each pair of vertices in this subset are connected by an edge. A maximum clique of a graph is a clique with the largest possible number of vertices for that graph.

APPENDIX B

Stochastic sub-gradient optimization based SVM

Here we describe the mathematical formulation behind the development of an algorithm for a stochastic sub-gradient solution to the optimization problem given in Equation (13).

Similar to gradient descent techniques, sub-gradient methods also operate in an iterative manner to reach an optimal or near-optimal solution to the optimization problem. Instead of taking a step in the direction opposite of the gradient as in gradient descent techniques, sub-gradient methods take a step in the direction opposite to the sub-gradient. The use of the sub-gradient instead of the gradient allows for handling the non-differentiability of the objective function but, at the same time, it implies that the value of the objective function can, and often does, increase over consecutive steps in sub-gradient methods. As a consequence, sub-gradient methods are not 'descent' methods and it is not possible to define a precise convergence criterion when using these methods. What is guaranteed for sub-gradient methods is that the distance to the set of near optimal solutions does not increase over iterations of the method.

In stochastic sub-gradient methods, at each iteration $t = 1, \ldots, T$ the sub-gradient $\boldsymbol{\nabla}_t$ with respect to the optimization variable ($\boldsymbol{w}$ in our case) is defined using a single randomly selected example instead of all examples. This allows for faster computation. In the case of our objective function, we define the sub-gradient $\boldsymbol{\nabla}_t$ at iteration t using a randomly selected classification example i_t from S, a randomly selected constraint (j_t, k_t) from D^{AB}, and a residue from each protein ($a_t \in A, b_t \in B$). The weight update equation can then be written as: $\boldsymbol{w}_{t+1} = \boldsymbol{w}_t - \eta_t \boldsymbol{\nabla}_t$ where the step size is given by: $\eta_t = \frac{1}{\lambda t}$. This choice of the step size improves the convergence bounds of the method as explained in [148]. Formally,

the sub-gradient for the objective function of Equation (13) can be written as:

$$\begin{aligned}\boldsymbol{\nabla}_t = \lambda \boldsymbol{w}_t - & \mathbb{1}\left(z_{i_t} < 1\right)\left\{\boldsymbol{\phi}\left(\boldsymbol{x}_{i_t}, y_{i_t}\right) - \boldsymbol{\phi}\left(\boldsymbol{x}_i, -y_i\right)\right\} - \\ & \nu \mathbb{1}\left(z_{(j_t,k_t)} < 1\right)\left\{\left(\boldsymbol{\phi}\left(\boldsymbol{x}^\star_{j_t}, y^{\star^t}_{j_t}\right) + \boldsymbol{\phi}\left(\boldsymbol{x}^\star_{k_t}, y^{\star^t}_{k_t}\right)\right) - \left(\boldsymbol{\phi}\left(\boldsymbol{x}^\star_{j_t}, 1\right) + \boldsymbol{\phi}\left(\boldsymbol{x}^\star_{k_t}, 1\right)\right)\right\} + \\ & \mu \mathbb{1}\left(z_{a_t} > c\right) \sum_{i \in E_A(a_t)} \left(\boldsymbol{\phi}\left(\boldsymbol{x}_i, +1\right) - \boldsymbol{\phi}\left(\boldsymbol{x}_i, -1\right)\right) + \\ & \mu \mathbb{1}\left(z_{b_t} > c\right) \sum_{i \in E_B(b_t)} \left(\boldsymbol{\phi}\left(\boldsymbol{x}_i, +1\right) - \boldsymbol{\phi}\left(\boldsymbol{x}_i, -1\right)\right).\end{aligned}$$

Here $\left(y^{\star^t}_{j_t}, y^{\star^t}_{k_t}\right) = \arg\max_{(y,y') \in V}\left(\boldsymbol{w}_t^T \boldsymbol{\phi}\left(\boldsymbol{x}^\star_{j_t}, y\right) + \boldsymbol{w}_t^T \boldsymbol{\phi}\left(\boldsymbol{x}_{k_t}, y'\right)\right)$. If $\boldsymbol{w}_0 = 0$, then, with the example i_t and constraint (j_t, k_t), the weight vector at iteration $t+1$ can be expressed as:

$$\begin{aligned}\boldsymbol{w}_{t+1} = & \frac{1}{\lambda t} \sum_{i=1}^{t} \mathbb{1}\left(z_{i_t} < 1\right)\left\{\boldsymbol{\phi}\left(\boldsymbol{x}_{i_t}, y_{i_t}\right) - \boldsymbol{\phi}\left(\boldsymbol{x}_{i_t}, -y_{i_t}\right)\right\} + \\ & \frac{\nu}{\lambda t} \sum_{i=1}^{t} \mathbb{1}\left(z_{(j_t,k_t)} < 1\right)\left\{\left(\boldsymbol{\phi}\left(\boldsymbol{x}^\star_{j_t}, y^{\star^t}_{j_t}\right) + \boldsymbol{\phi}\left(\boldsymbol{x}^\star_{k_t}, y^{\star^t}_{k_t}\right)\right) - \left(\boldsymbol{\phi}\left(\boldsymbol{x}^\star_{j_t}, 1\right) + \boldsymbol{\phi}\left(\boldsymbol{x}^\star_{k_t}, 1\right)\right)\right\} - \\ & \frac{\mu}{\lambda t} \sum_{j=1}^{t} \mathbb{1}\left(z_{a_t} > c\right) \sum_{i \in E_A(a_t)} \left(\boldsymbol{\phi}\left(\boldsymbol{x}_i, +1\right) - \boldsymbol{\phi}\left(\boldsymbol{x}_i, -1\right)\right) - \\ & \frac{\mu}{\lambda t} \sum_{j=1}^{t} \mathbb{1}\left(z_{b_t} > c\right) \sum_{i \in E_B(b_t)} \left(\boldsymbol{\phi}\left(\boldsymbol{x}_i, +1\right) - \boldsymbol{\phi}\left(\boldsymbol{x}_i, -1\right)\right).\end{aligned}$$

Note that, in the above equation, each term includes a count of the number of times a constraint has been violated up to the current iteration $t+1$. If we use $\alpha_{t+1}[i]$, $\alpha_{t+1}[j,k]$, $\alpha^A_{t+1}[a]$ and $\alpha^B_{t+1}[b]$ to indicate the number of times, up to the current iteration $t+1$, $z_i < 1$, $z_{(j,k)} < 1$, $z_a > c$ and $z_b > c$ respectively, then we can simplify the above equation as follows.

$$\begin{aligned}\boldsymbol{w}_{t+1} =& \frac{1}{\lambda t}\sum_{i=1}^{N}\alpha_{t+1}\left[i\right]\left\{\boldsymbol{\phi}\left(\boldsymbol{x}_i, y_i\right) - \boldsymbol{\phi}\left(\boldsymbol{x}_i, -y_i\right)\right\} + \\ & \frac{\nu}{\lambda t}\sum_{(j,k)\in D^{AB}}\alpha_{t+1}\left[j,k\right]\left\{\left(\boldsymbol{\phi}\left(\boldsymbol{x}_j^{\star}, y_j^{\star^t}\right) + \boldsymbol{\phi}\left(\boldsymbol{x}_k^{\star}, y_k^{\star^t}\right)\right) - \left(\boldsymbol{\phi}\left(\boldsymbol{x}_j^{\star}, 1\right) + \boldsymbol{\phi}\left(\boldsymbol{x}_k^{\star}, 1\right)\right)\right\} - \\ & \frac{\mu}{\lambda t}\sum_{a\in A}\alpha_{t+1}^{A}\left[a\right]\sum_{i\in E_A(a)}\left(\boldsymbol{\phi}\left(\boldsymbol{x}_i, +1\right) - \boldsymbol{\phi}\left(\boldsymbol{x}_i, -1\right)\right) - \\ & \frac{\mu}{\lambda t}\sum_{b\in B}\alpha_{t+1}^{B}\left[b\right]\sum_{i\in E_B(b)}\left(\boldsymbol{\phi}\left(\boldsymbol{x}_i, +1\right) - \boldsymbol{\phi}\left(\boldsymbol{x}_i, -1\right)\right).\end{aligned}$$

Now, the output $f_{t+1}\left(\boldsymbol{x}, y\right) = \left\langle \boldsymbol{w}_{t+1}, \phi\left(\boldsymbol{x}, y\right)\right\rangle$ can be written as:

$$\begin{aligned}f_{t+1}\left(\boldsymbol{x}, y\right) =& \frac{1}{\lambda t}\sum_{i=1}^{N}\alpha_{t+1}\left[i\right]\left\{K\left(\boldsymbol{x}_i, y_i, \boldsymbol{x}, y\right) - K\left(\boldsymbol{x}_i, -y_i, \boldsymbol{x}, y\right)\right\} + \\ & \frac{\nu}{\lambda t}\sum_{(j,k)\in D^{AB}}\alpha_{t+1}\left[j,k\right]\left(K\left(\left(\boldsymbol{x}_j^{\star}, y_j^{\star^t}\right), \left(\boldsymbol{x}, y\right)\right) + K\left(\left(\boldsymbol{x}_k^{\star}, y_k^{\star^t}\right), \left(\boldsymbol{x}, y\right)\right)\right) - \\ & \frac{\nu}{\lambda t}\sum_{(j,k)\in D^{AB}}\alpha_{t+1}\left[j,k\right]\left(K\left(\left(\boldsymbol{x}_j^{\star}, 1\right), \left(\boldsymbol{x}, y\right)\right) + K\left(\left(\boldsymbol{x}_k^{\star}, 1\right), \left(\boldsymbol{x}, y\right)\right)\right)\} - \\ & \frac{\mu}{\lambda t}\sum_{a\in A}\alpha_{t+1}^{A}\left[a\right]\left\{\sum_{i\in E_A(a)}K\left(\boldsymbol{x}_i, +1, x, y\right) - K\left(\boldsymbol{x}_i, -1, x, y\right)\right\} - \\ & \frac{\mu}{\lambda t}\sum_{b\in B}\alpha_{t+1}^{B}\left[b\right]\left\{\sum_{i\in E_B(b)}K\left(\boldsymbol{x}_i, +1, x, y\right) - K\left(\boldsymbol{x}_i, +1, x, y\right)\right\}.\end{aligned}$$

Based on the feature representation $\boldsymbol{\phi}\left(\boldsymbol{x}, y\right)$, the kernel $K\left(\boldsymbol{x}, y, \boldsymbol{x}', y'\right) = \left\langle \phi\left(\boldsymbol{x}, y\right), \phi\left(\boldsymbol{x}', y'\right)\right\rangle$ can be written as: $K\left(\left(\boldsymbol{x}, y\right), \left(\boldsymbol{x}', y'\right)\right) = K_{pw}\left(\boldsymbol{x}, \boldsymbol{x}'\right)\mathbb{1}\left(y = y'\right)$ with K_{pw} being the pairwise kernel combination used in PAIRpred. Thus, $f\left(\boldsymbol{x}, y\right)$ can be expressed as follows.

$$
\begin{aligned}
f_{t+1}(\boldsymbol{x}, y) =& \frac{1}{\lambda t} \sum_{i=1}^{N} \alpha_{t+1}[i] \left\{ K_{pw}(\boldsymbol{x}_i, \boldsymbol{x}) \left(\mathbb{1}(y = y_i) - \mathbb{1}(y = -y_i) \right) \right\} + \\
& \frac{1}{\lambda t} \sum_{(j,k) \in D^{AB}} \alpha_{t+1}[j,k] K_{pw}\left(\boldsymbol{x}_j^{\star}, \boldsymbol{x}\right) \left(\mathbb{1}\left(y = y_j^{\star^t}\right) - \mathbb{1}(y = 1) \right) + \\
& \frac{1}{\lambda t} \sum_{(j,k) \in D^{AB}} \alpha_{t+1}[j,k] K_{pw}(\boldsymbol{x}_k^{\star}, \boldsymbol{x}) \left(\mathbb{1}\left(y = y_k^{\star^t}\right) - \mathbb{1}(y = 1) \right) - \\
& \frac{\mu}{\lambda t} \sum_{a \in A} \alpha_{t+1}^{A}[a] \left\{ \sum_{i \in E_A(a)} K(\boldsymbol{x}_i, x) \left(\mathbb{1}(y = 1) - \mathbb{1}(y = -1) \right) \right\} - \\
& \frac{\mu}{\lambda t} \sum_{b \in B} \alpha_{t+1}^{B}[b] \left\{ \sum_{i \in E_B(b)} K(\boldsymbol{x}_i, x) \left(\mathbb{1}(y = 1) - \mathbb{1}(y = -1) \right) \right\}.
\end{aligned}
$$

Since, $\mathbb{1}(y = y_i) - \mathbb{1}(y = -y_i) = y y_i$, $\mathbb{1}\left(y = y_j^{\star}\right) - \mathbb{1}(y = 1) = \dfrac{y(y_j^{\star} - 1)}{2}$ and $\mathbb{1}(y = 1) - \mathbb{1}(y = -1) = y$, $f_{t+1}(\boldsymbol{x}, y)$ in the above equation can be written as: $f_{t+1}(\boldsymbol{x}, y) = y f'_{t+1}(\boldsymbol{x})$, where $f'_{t+1}(\boldsymbol{x})$ is given by:

$$
\begin{aligned}
(22) \quad f'_{t+1}(\boldsymbol{x}) =& \frac{1}{\lambda t} \sum_{i=1}^{N} \alpha_{t+1}[i] \left\{ K_{pw}(\boldsymbol{x}_i, \boldsymbol{x}) y_i \right\} + \\
& \frac{\nu}{2\lambda t} \sum_{(j,k) \in D^{AB}} \alpha_{t+1}[j,k] \left\{ K_{pw}\left(\boldsymbol{x}_j^{\star}, \boldsymbol{x}\right) \left(y_j^{\star^t} - 1 \right) + K_{pw}(\boldsymbol{x}_k^{\star}, \boldsymbol{x}) \left(y_k^{\star^t} - 1 \right) \right\} - \\
& \frac{\mu}{\lambda t} \sum_{a \in A} \alpha_{t+1}^{A}[a] \sum_{i \in E_A(a)} K(\boldsymbol{x}_i, \boldsymbol{x}) - \frac{\mu}{\lambda t} \sum_{b \in B} \alpha_{t+1}^{B}[b] \sum_{i \in E_B(b)} K(\boldsymbol{x}_i, \boldsymbol{x})
\end{aligned}
$$

Recall that, for each iteration $t+1$, $\left(y_j^{\star^t}, y_k^{\star^t}\right) = \arg\max_{(y,y') \in V} \left(f_t\left(\boldsymbol{x}_j^{\star}, y\right) + f_t(\boldsymbol{x}_k^{\star}, y') \right)$. The scoring function for a test example $\boldsymbol{x}$ after training is then computed as: $f(\boldsymbol{x}) = f_T(\boldsymbol{x}, +1) - f_T(\boldsymbol{x}, -1)$, which should be positive for examples in the test set that interact and negative otherwise. Please note that, even though the weight vector and the output function have been

expressed solely in terms of kernel operators, the problem is still being solved in the primal using the sub-gradient over the weight vector. This is because the objective function, though convex with respect to the weight vector, may not be strongly convex with respect to $\boldsymbol{\alpha}$ [148]. This algorithm is guaranteed to converge to an $\epsilon > 0$ accurate solution within $O\left(\frac{R^2}{\lambda\epsilon}\right)$ iterations, where $R = 2\max_{\boldsymbol{x}}\left(\|\Phi\left(\boldsymbol{x}\right)\|\right)$ and $\Phi\left(\boldsymbol{x}\right)$ is the implicit feature representation from the kernel K_{pw} of an example in the training and test set[14]. This shows that the number of iterations is independent of the number of examples. Here, we also present a pseudo-code of the algorithm that can be used to find $\alpha\left[\cdot\right]$ and $\alpha\left[\cdot,\cdot\right]$. Algorithm 4 is inspired from the kernelized Pegasos algorithm [148].

[14]The proof for this bound is omitted since it is similar to the ones given in [145, 148]

Algorithm 4 SSO for SVM with pairwise constraints

Input: S: Labeled training data, $S^\star$: Test samples with unknown labels, D^{AB}: List of index tuples (j,k) from $S^\star$ that define pairwise constraints over $S^\star$.

Parameters: $\lambda \geq 0$: Regularization parameter, $T > 0$: Number of iterations, $0 \leq \nu \leq 1$: Controls relative contribution from the training data and pairwise constraints, $0 \leq \mu \leq 1$: Controls relative contribution from the sparsity constraints.

Output: Prediction scores $z_j^\star$ and labels $y_j^\star$ for all test examples $j^\star \in S^\star$.

Set $N =$ Number of training examples in S.
Set $M =$ Length of D^{AB}.
Set $\alpha_0[j] = 0, \forall j = 1, \ldots N$.
Set $\alpha_0[j,k] = 0, \forall (j,k) \in D^{AB}$.
Set $\alpha_0^A[j] = 0, \forall j \in A$.
Set $\alpha_0^B[j] = 0, \forall j \in B$.
Set $y_j^\star = -1, \forall j \in S^\star$.
for $t = 0, \ldots, T-1$ **do**
  Set $\alpha_{t+1}[j] = \alpha_t[j], \forall j = 1, \ldots N$.
  Set $\alpha_{t+1}[j,k] = \alpha_t[j,k], \forall (j,k) \in D^{AB}$.
  Set $\alpha_{t+1}^A[j] = \alpha_t^A[j], \forall j \in A$.
  Set $\alpha_{t+1}^B[j] = \alpha_t^B[j], \forall j \in B$.
  Select a training example i_t from S randomly.
  if $y_{i_t} f_t'(\boldsymbol{x}_{i_t}) < 1$ **then**
    $\alpha_{t+1}[i_t] = \alpha_t[i_t] + 1$.
  end if
  Select (j,k) from D^{AB} randomly.
  Set $(y_j^\star, y_k^\star) = \arg\max_{(y,y') \in V} \left(y f_{t+1}'(\boldsymbol{x}_j^\star) + y' f_{t+1}'(\boldsymbol{x}_k^\star) \right)$
  if $\left((y_j^\star - 1) f_{t+1}'(\boldsymbol{x}_j^\star) + (y_k^\star - 1) f_{t+1}'(\boldsymbol{x}_k^\star) \right) < 1$ **then**
    $\alpha_{t+1}[j,k] = \alpha_t[j,k] + 1$.
  end if
  Select a residue a from A randomly and compute $z_a = \frac{1}{\mid E_A(a) \mid} \sum_{i \in E_A(a)} f_{t+1}(\boldsymbol{x}_i^\star)$
  if $z_a > c$ **then**
    $\alpha_{t+1}^A[a] = \alpha_t^A[a] + 1$
  end if
  Select a residue b from B randomly and compute $z_b = \frac{1}{\mid E_B(b) \mid} \sum_{i \in E_B(b)} f_{t+1}(\boldsymbol{x}_i^\star)$
  if $z_b > c$ **then**
    $\alpha_{t+1}^B[b] = \alpha_t^B[b] + 1$
  end if
end for
Use Equation 22 for $t = T$ together with $\alpha_T[\cdot]$, $\alpha_T[\cdot,\cdot]$ and the test labels $y^\star$ obtained above to compute the output scores and labels for all examples in $S^\star$.