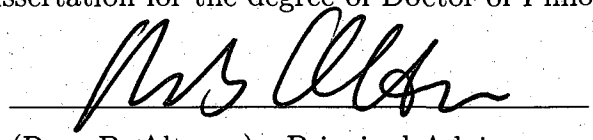
CHARACTERIZATION OF PROTEIN FUNCTION USING AUTOMATED COMPUTATIONAL METHODS

A DISSERTATION
SUBMITTED TO THE DEPARTMENT OF BIOMEDICAL INFORMATICS
AND THE COMMITTEE ON GRADUATE STUDIES
OF STANFORD UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Shirley Wu

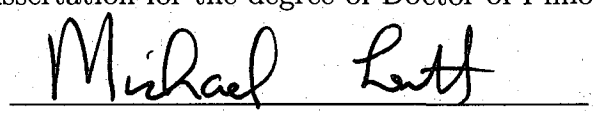
June 2009

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.



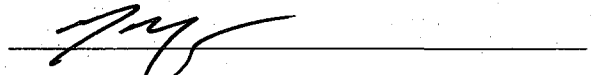
(Russ B. Altman) Principal Advisor

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.



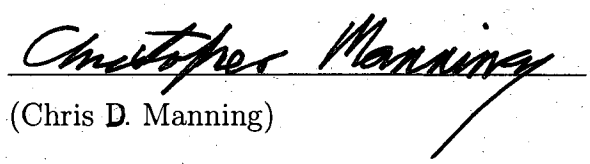
(Michael Levitt)

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.



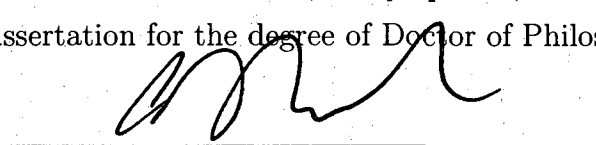
(Serafim Batzoglou)

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.



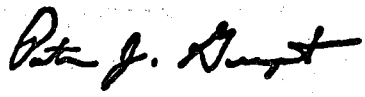
(Chris D. Manning)

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.



(Gavin Sherlock)

Approved for the University Committee on Graduate Studies.



Abstract

The popularization of high-throughput biological techniques has produced a significant bottleneck between protein identification and functional annotation. To alleviate this problem, researchers often apply computational methods for protein function recognition; however, existing tools are not as effective when the proteins are structurally novel. Structural genomics projects in particular are generating many novel protein structures with little associated functional knowledge, and so new function characterization methods that do not rely on strict sequence or structural similarity are needed. Thanks to improvements in sequencing technologies, we are also now discovering new proteins at a faster rate. These proteins may contain novel biological functions, but existing approaches are ill-equipped to discover them.

In this dissertation, I present several methods for protein function characterization that can be combined into pipelines both for supervised modeling of known functional sites and for unsupervised discovery of potentially novel functional sites. Each pipeline takes advantage of an existing framework called FEATURE, which models functional sites in protein structures. The first method, SeqFEATURE, uses sequence motifs to seed 3D models which are more robust to reductions in sequence identity compared to other sequence-based methods. The models are also more sensitive than other structure-based methods when tested on proteins with low structural similarity to

known proteins. Using SeqFEATURE, I created and validated a large library of 3D functional site models and scanned all structures in the Protein Data Bank with each model, including structures with unknown function from structural genomics projects. The data and models are publicly available.

To identify and characterize potentially novel biological functions, we combine a number of clustering techniques with knowledge-informed approaches. FEATURE generates descriptive vectors of protein microenvironments, which we cluster using k-means to identify environments that recur across different protein structures. Each cluster represents a potential biological site of interest, but is likely to be noisy and therefore difficult to interpret. To select candidate clusters for analysis, I used hierarchical clustering in conjunction with a scoring function that takes into account the functional and internal coherence of sub-clusters. To annotate resulting candidate clusters, I developed a set of methods for ranking important terms found in the literature and in database records associated with the proteins comprising the cluster. We applied these methods to a novel data set of cysteine-based protein microenvironments, rediscovering known functional sites and sub-classes of functional sites in addition to making several novel predictions.

This dissertation extends existing frameworks to be relevant in the context of structural genomics. I demonstrate and validate an approach for rapid creation of robust functional site models that can be applied in high-throughput, and define a pipeline by which novel biology can be discovered and characterized. The work presented demonstrates significant contributions towards the characterization of protein function – both known and novel – using computational methods.

Acknowledgments

It's hard to believe that five years have gone by. In that time, I have learned so much from family, friends, colleagues, and mentors. Not just academic lessons – like how to evaluate a classifier or write a paper – but also everyday wisdom. I've learned that teams work better together when they enjoy one another's company and have mutual respect. I've learned that good food is not only filling, but fulfilling. Most importantly, I've learned that life is best when you surround yourself with happy, active people with diverse interests who appreciate simple things like the grass on the hills or the sun on the water. Stanford and the Bay area are full of these types of people and I am thankful to have had the opportunity to come here.

My learning in some sense is epitomized by this dissertation but it has been a group effort from the start. It began where a previous graduate student, Mike Liang, left off and has subsequently been supported by many others – Inbal Landsberg, Jessica Ebert, Dariya Glazer, Tianyun Liu, and Megan So – who have worked on related projects. My work on SeqFEATURE is a direct extension of Mike's dissertation, and I would not have had any clustering data to apply my methods to without Tianyun's help. I also thank my many advisors and mentors for their guidance as I navigated my way through coursework, research projects, job searches, proposals, and presentations. In particular, I thank my thesis committee members, Michael Levitt,

Serafim Batzoglou, Gavin Sherlock, and Christopher Manning, for providing me with constructive feedback that never failed to improve my work or make me think; my academic advisor, Teri Klein, for sage and pointed advice on life and careers; Douglas Brutlag for advising me through one of my first year rotations, serving as the Chair of my Orals committee, and always being an engaged member of the BMI community; Larry Fagan and Betty Cheng for their invaluable assistance with talks, especially my oral defense; and the rest of the BMI exec for their tireless efforts in making the Biomedical Informatics program at Stanford so unique.

Special thanks go to my primary advisor, Russ Altman, who is his own breed of professor. Who else will dispense critical feedback while cracking a joke, shepherd students on 100 mile bike rides to Monterey, or pretend to be Diana Ross with such enthusiasm? All this while heading multiple million-dollar grants, directing multiple departments, and serving as advisor or member of numerous boards and committees. He has been extremely supportive of my extracurricular activities, letting my Ultimate Frisbee team take over his house for a day to hold a fund-raising garage sale and taking seriously my suggestion to start blogging. His abundant energy is always positive and infectious.

Russ has shown me through his example that it is possible to be a top tier scientist, manager, and mentor – qualities that can often seem mutually exclusive – without sacrificing family or hobby, and that one can combine many interests into a satisfying career. He listened without judgment when I contemplated leaving graduate school after two years and told me simply to do what would make me happy. It is clear, from seeing his mentorship of other students and experiencing it first hand, that his goal is to help us be successful in whatever way is best for us. Russ is truly an inspirational figure and it has been a privilege to be his student.

My other colleagues in the Altman lab have been there through it all – thorny research problems, the mid-Ph.D. crisis, failed fellowship applications, successful proposals, and epic karaoke acts. Who else would even attempt, let alone pull off "We Are the Nerds" or "Bioinformatician Rhapsody?" I couldn't ask for better labmates than Magda Jonikas, Bernie Daigle, Yael Garten, and Alain Laederach, and I'll always remember the movie nights (including Donnie Darko in S222), lunches at Thai Cafe, jogging in Menlo Park while Alain rowed his bike, and the many talks we had about work and life with fondness. Thanks also go to Randy Radmer for helping us with myriad technical problems and others for making the Altman lab such a fun, intellectually stimulating, and supportive place to be a graduate student.

To my friends in BMI, you made my graduate experience a thoroughly enjoyable one. Runa Islam, Sam Pearlman, Amit Kaushal, Kaustubh Supekar, Rong Xu, Chris "CJ" Johnson, and Roni Zeiger have been great classmates (though I use the term "classmates" loosely here), some hilariously so. No one embraced the 80's for our rendition of "Tainted Love" at the BMI retreat in 2007 more than Runa and CJ! I am grateful to those who graduated before me – Zach Pincus, Brian Naughton, Serge Saxonov, Jessie Tenenbaum, Nikesh Kotecha, Lucy Southworth, Maureen Hillenmeyer, and others – for being great mentors and role models. Those who have yet to graduate – Sarah Aerni, Tiffany Chen, David Chen, Alex Morgan, Noah Zimmerman, Wei-chih Lee, Jesse Rodriguez, Marina Sirota, and others – have made BMI an even more fun place. Behind the scenes, Carol Maxwell, Darlene Vian, Christine Hilliard, Beth McKeown, Mary Jeanne Oliva, and others have all worked tirelessly – and always with a smile – to make BMI run smoothly and I cannot thank them enough. Tiffany Murray, who has helped Russ for many years, has also given me invaluable assistance, especially with running a workshop at PSB.

Through blogging and FriendFeed I have met a vibrant community of scientists who continue to impress me with their passion for their work, their willingness to engage and help others, and their commitment to improving science. Cameron Neylon, Neil Saunders, Bora Zivkovic, Jean-Claude Bradley, Pedro Beltrao, Jim Hardy, Deepak Singh, and others encouraged my foray into open science and science communication, leading to opportunities organizing the PSB workshop on open science and, I'm convinced, landing me a position at an innovative science start-up where I can potentially merge my varied skills and interests. Iddo Friedberg, a colleague in the area of protein function prediction, has also given me much encouragement in my recent endeavors.

To my oldest friends, thank you for your continued support and friendship. Selena Liao, Nancy Sun, Albert Shiue, Stan Su, Matthew Bills, Cathy Cheung, Keiko Ono, and Kevin Wong, I know that whenever I need something, I can call on you to help. To Eugene Song, for sharing those "wtf" moments and never failing to make me laugh (or shake my fist). To kindred spirits, for seeing all that is beautiful, nurturing the artist, and keeping the wonder alive. To all of my friends from Brown – on Disco, on B-Mo, the class of 2004, and others – I have friends wherever I go due in large part to you.

Outside of Stanford, I have had the good fortune to become friends with many amazing, intelligent people. Wes and Gillian Chao, Mark and Megan Smith, Kris McQueen and Kitt Hodsden, Keith and Katie Randall, David and Nichole Pickett, Warren Schechter and Megan Donahue, Paul Youn, Steffi Wu, Will Goodyer, Mikael "Chookie" Arneborn and Martha Montague, Andy Crews, Wade and Christina Hellner, and others on Mischief, in the Ultimate community, and beyond, thanks so much for being my teammates, for the communal dinners and game nights, and the pleasure

of your company. Thanks to all the kids – Mirabelle, Mayanna, Alex "Danger", Mia "Dare", Sophie, Adrian, Linnea, Sugar, and Biscuit – for being cute and reminding me what a joy life and learning can be. To Chris Doyle, who has been a constant, steadying force for me these last few years, thank you for all the good cooking, the debugging sessions, and, most importantly, your love and support.

And, of course, I would not be here without my family. Thank you to my brothers, Wayne and Steven, for supporting me in your own special way, and for the fact that we can still fall into fits of laughter without saying anything. I especially dedicate this dissertation to my parents, who are a shining example of what you can accomplish with hard work and determination. At an early age, they instilled in me a similar work ethic and a love for education of all kinds, not just the kind you find in books but also wherever the possibility presents itself. As I grow older, I appreciate everything my parents did for us and taught us more and more. Because of them, I am an adventurous eater, an avid reader, and I am in constant wonder of nature and life on this planet. Perhaps most importantly, my brothers and I never lacked for opportunities to try what we were interested in and my parents demanded only that we do what we love while being good citizens of the world. For that, I will always be in their debt.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Knowledge of protein function is essential for understanding biological processes, and is important for treating disease and engineering beneficial outputs such as biofuels [171, 134]. Detailed knowledge of function – and, increasingly, structure – is especially relevant for drug development since specific targeting of proteins based on these data helps to increase efficacy and reduce side effects [42, 35]. Information about the function of proteins is usually deduced through biological assays probing their expression and regulation, cellular localization, and interaction partners, among other data. This is often a trial and error process, and so is extremely time and resource-intensive.

Motivating automated protein function annotation

With the advent of high-throughput technology, we now have many proteins lacking functional annotation, and it is clear that manual annotation efforts are insufficient [13]. At first, the flood came from protein sequences arising from the many genome projects. The 2001 Protein Structure Initiative (PSI) [91], however, spurred technological advances in structure determination, and solving protein structures has now also become a high-throughput endeavor [23, 27, 103]. This fact, combined with the

overall goal of structural genomics (SG) of enhancing coverage of structure space, has resulted in the rise of a new class of proteins: those with solved structures but virtually no functional information (see Figure 1.1) [62, 93]. Given the difficulty of assaying function experimentally, computational methods for function prediction are necessary to provide preliminary annotations and to guide functional studies.

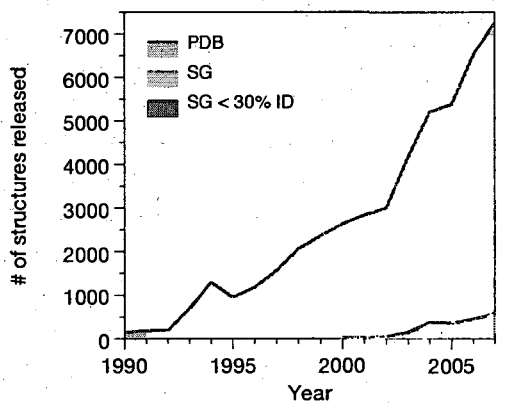


Figure 1.1: Proteins from structural genomics projects in the Protein Data Bank (PDB). The number of protein structures released in the PDB [15] each year is increasing, as is the number of proteins in the PDB that are from SG centers. The number of novel proteins solved by SG centers comprises a significant portion (about 40%) of the structures released. Data from the PDB and TargetDB websites [117, 148].

Numerous tools exist for predicting function using both sequence and structure, and almost all rely on the similarity of the query protein to known sequences or structures. These methods tend to have good performance when homology is present and the matching proteins are well-characterized, but they are less helpful when the structure – and, therefore, sequence – is novel [81]. A study examining the usefulness of a suite of sequence and structure-based tools for predicting function on structural genomics targets found not only that structure-based tools were most successful, but that no one method was always successful [159]. This underscores the importance of structure-based methods for function prediction as well as the need for different and complementary tools [133].

In 2003, the PSI announced supplemental grants for functional studies on structural genomics targets [142], acknowledging that a structure with characterized function is more desirable than an uncharacterized structure. Given the emphasis placed

on elucidating the function of these thousands of unannotated protein structures, it is important to develop and make available tools that allow efficient and comprehensive scanning of function on a large scale, and provide intuitive interpretations of the resulting predictions. Another consideration is the possibility that the proteins possess functions that have not previously been seen. Existing function prediction methods are predominantly built for known functions and require training sets of examples. There is thus a need to develop methods for discovering novel protein functions so that we can model them for recognition tasks.

This dissertation builds largely upon an existing framework for modeling functional sites in protein structures, called FEATURE [162, 59]. Rather than using strict sequence or structure matching, it represents sites as a sphere of physical and chemical properties derived from the structure [8]. FEATURE is flexible and intuitive, but historically was not well-suited for analyzing structural genomics proteins because of the scarcity of models. My dissertation work focuses on extending the FEATURE framework in two ways. One is by extending and validating an approach for rapid construction of robust functional site models that can be applied in high-throughput to structural genomics targets. The other defines a pipeline through which previously unknown biological functions can be discovered and characterized.

Automated generation of 3D models from 1D motifs

In the first part of this dissertation, I describe an approach that builds upon earlier work [95], called SeqFEATURE, which allows automatic generation of training sets from sequence motifs for use in defining models. We have used SeqFEATURE to construct a large library of 136 functional site models and have validated it internally as well as through a comparison to existing sequence and structure-based methods

[165]. In particular, SeqFEATURE models are more robust than other methods when the query protein exhibits low sequence and structural similarity to known proteins (Figure 1.2). As sequence identity is reduced, SeqFEATURE's sensitivity stays constant, while the sensitivity of sequence-based methods exhibits a definitive decline. Similarly, SeqFEATURE maintains relatively high sensitivity when tested on proteins with low structural similarity to known proteins, compared to the best performing structure-based method. We have used the library to scan the entire PDB [136], including structures in the TargetDB repository for structural genomics targets [29], and have made the data available through a web server, called WebFEATURE [166]. Users may also scan structures of interest with all of the models in the SeqFEATURE library and interactively view results through WebFEATURE.

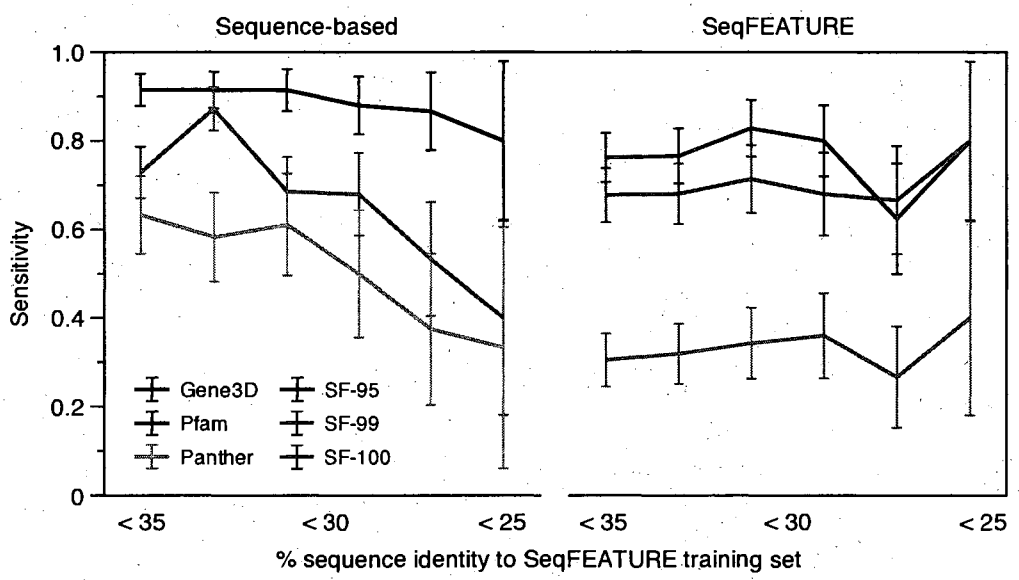


Figure 1.2: Performance of sequence-based methods compared to SeqFEATURE at low sequence identity. Sensitivity of the three sequence-based methods decreases as the sequence identity of the query to known proteins is reduced. In contrast, the sensitivity of SeqFEATURE (at three different specificity-based score cutoffs) remains robust to sequence identity. *Note: this is also Figure 3.6.*

Towards a protein function recognition pipeline

In addition to creating a comprehensive and well-validated library of 3D models for function prediction, I have explored the utility of the FEATURE framework to discover and model potentially novel functions. A previous study clustered microenvironments over a non-redundant subset of the PDB [169], and we have developed more effective clustering and cluster analysis methods to produce clusters that are biologically significant and more easily interpretable. I contribute to this work in two ways: by adapting and developing methods to identify smaller, functionally coherent sub-clusters from larger, coarse-grained clustering results; and by incorporating knowledge from databases and scientific literature to generate detailed annotations.

To prioritize and refine protein clusters for annotation, I have adapted the neighbor divergence per gene (NDPG) [129, 130] algorithm which determines the functional coherence of clusters. Our tests indicate that functional protein clusters have much greater functional coherence than completely random clusters, and that functional coherence decreases with the amount of functional signal in the cluster. Using hierarchical clustering with a scoring function combining functional coherence, internal coherence, and cluster size allows us to refine a large cluster of protein microenvironments into smaller, more coherent sub-clusters.

To help characterize the resulting sub-clusters, I have incorporated knowledge from literature and other databases to produce ranked lists of terms. To score and rank potential literature terms, we employ scoring functions based on the hypergeometric distribution as well as the concept of entropy from information theory. Database terms are scored according to the hypergeometric distribution only. We present the top ranked terms in a summary HTML page containing links to more detailed information for each term category, including the proteins that contributed to each term.

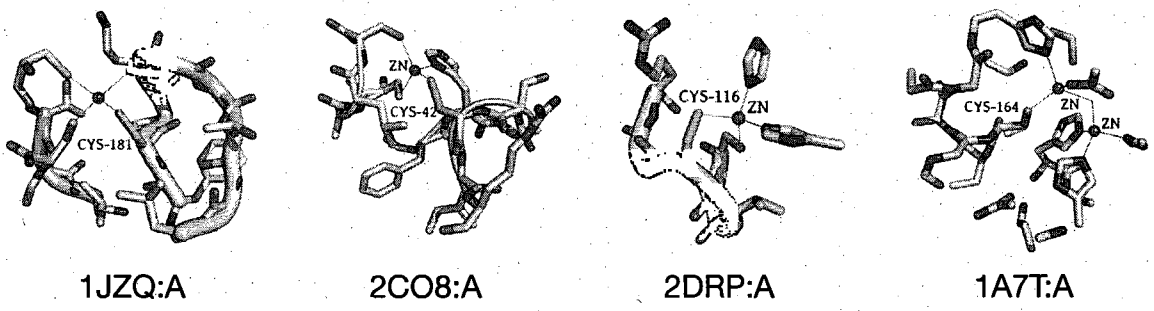


Figure 1.3: Four distinct sub-classes of zinc binding sites. Using unsupervised clustering techniques in combination with literature-based functional coherence filtering, we are able not only to rediscover zinc binding sites, but to distinguish between sub-types of zinc binding. Here, we show representative microenvironments from four distinct zinc binding sub-clusters. *Note: this is also Figure 5.11.*

Using these methods, we identify a number of clusters that recapitulate known functions and show that we can distinguish sub-classes of sites with similar functions (see Figure 1.3). In addition, we present intriguing examples of potentially new functional sites, including novel annotations for individual proteins and entire sub-clusters that may represent novel motifs or functions. Annotated clusters can then be used to train additional functional models to expand the existing library, creating an iterative pipeline for discovering and modeling protein functional sites.

In the remainder of this document, I review the background relevant to my work in more detail in Chapters 2 and 4, present the methods and results for the Seq-FEATURE study in Chapter 3, and describe the methods I have developed for cluster prioritization and annotation and results of their application in Chapter 5. I discuss the implications and contributions of this work as well as future research directions in Chapter 6.

Chapter 2

A review of protein function prediction

Protein function prediction is a multi-faceted problem and many different approaches exist. The most obvious distinction between methods arises from the type of information used to model the function; for our purposes we will consider the two most common and direct forms, sequence data and structure data. Another difference is the granularity of the function modeled – e.g. the method may produce an annotation to a biological process, classification into a protein family, labeling of sub-domains, or identification of specific binding sites. Finally, we can contrast the methods themselves based on the algorithms used.

Most methods perform well under specific circumstances but few can be applied with good results in all situations, making a diversity of tools desirable [133]. As

Portions of this chapter have appeared in the following papers: Wu S, Liang MP, Altman RB. (2008) The SeqFEATURE library of 3D functional site models: comparison to existing methods and applications to protein function annotation. Genome Biology 9:R8; and Halperin I*, Glazer DS*, Wu S*, Altman RB. (2008) The FEATURE framework for protein function annotation: modeling new functions, improving performance, and extending to novel applications. BMC Genomics 9(Suppl 2):S2. **Contributed equally.*

technology continues to improve and the rate of protein discovery increases, function annotation tools that make high-quality predictions in high-throughput will become a necessity. This chapter will discuss a representative sample of the sequence and structure-based function prediction tools available, and, in particular, will describe a versatile framework for modeling functional sites in protein structures, called FEATURE, which provides the foundation for this dissertation work.

2.1 Sequence-based function prediction methods

The majority of predictors use primary sequence, and the simplest method is to use a sequence alignment algorithm such as BLAST [3], since high sequence similarity is almost always indicative of evolutionary – and, therefore, functional – conservation. Wilson *et al.* [163] showed that precise function can be transferred reliably above 40% and broad functional class above 25% sequence identity. New proteins can thus theoretically be annotated with the functions of their close sequence neighbors. In addition, many tools take advantage of curated databases, such as the manually inspected profile-Hidden Markov Models (HMMs) contained in the Pfam database of protein families [141], and PROSITE, which consists of manually curated sequence patterns and profiles [69]. Many functions such as binding sites or enzyme active sites are conserved in sequence, and sequence motifs like the ones above can be used to detect them in new sequences.

Both Pfam and PROSITE are contained within InterPro [71, 109], a comprehensive, integrated resource for protein sequence information that provides many databases and tools for protein function and domain recognition. Among the tools

offered are other HMM-based methods [12] such as HMMTigr, built on the TIGRFAMs database [58], and HMMPanther, built on the PANTHER database [149], both of which focus on function-based classification. Superfamily [101], another HMM-based tool hosted on InterPro, classifies sequences using manually curated models built from the Structural Classification of Proteins (SCOP) [110]. As a complement to Superfamily, Gene3D [25] is a semi-manually curated set of models built using the CATH protein structure classification [57].

The collection of sequence-based functional motifs, domains, and families, and their accompanying prediction tools mentioned above are considered the state of the art in sequence-based function prediction. Given the ubiquity of sequence information, these tools perform extremely well under most circumstances. Because they rely on sequence similarity to characterized proteins or domains, however, they are typically unable to provide useful results for proteins lacking that similarity – a scenario that is becoming more and more common. Structure is known to be more conserved than sequence [31], so structure-based prediction methods are needed. These will be more effective even at sequence identities too low for reliable annotation transfer by sequence-based methods [159].

2.2 Structure-based function prediction methods

Sequence-based tools often provide useful information about function, but they may be less suited to cases where sequence identity is low. Under these circumstances, structure-based tools may detect functional signals that sequence-based methods are unable to capture due to sequence divergence [81]. Since a protein's structure and function are inexorably linked, structure-based tools can abstract out those elements

that are necessary for defining a particular function independent of the linear sequence, lending a degree of sensitivity and specificity that may improve over sequence-based tools. The abstractions can range in scale from entire secondary structure elements to residue or atom-based features. Function annotation based on structure is usually limited to recognition of either general folds or low-level molecular functions such as binding sites and active sites; it is unlikely routinely to predict the overall biological pathways and processes in which a protein participates. However, a complete understanding of structural environments and binding and active site properties provides a pyramid of evidence for the functional roles of a protein.

Protein structure is complex, so simplified representations are used to capture relevant features in a way that is computationally tractable. Methods such as CASTp [16] employ geometric abstractions to describe the shape, area, and volume of surface pockets and internal cavities, which are often correlated with functional sites. Geometry can also be used to determine the relative position of several amino acids to each other. Other representations involve calculating values for physicochemical properties associated with locations or elements in the structure, such as solvent accessibility, hydrophobicity, electrostatic potential, the presence of residues or secondary structure, conservation or the presence of chemical groups [59, 158, 86, 121, 170, 87]. Jambon *et al.* [75] use a representation that combines both geometry and property-based components.

Some methods for structure-based protein function prediction rely on expert knowledge for defining the features useful for classifying a particular functional site, while others learn the important features through supervised machine learning approaches. An example of the former is Fuzzy Functional Forms (FFFs) [45], which are three-dimensional descriptions of functional sites based on conserved geometry,

protein conformation, and residue identity. The descriptions are built by hand using information from solved crystal structures and published literature. FFF's were able to help identify functional sites in structures whose sequence similarity to known proteins was low enough to render sequence-based tools ineffective [46].

Constructing models manually is time-consuming, however, and several more tractable methods have since been developed. ProKnow [116] uses features extracted from sequence or structure via established tools such PSI-BLAST [4], DALI [66], PROSITE, and the Database of Interacting Proteins (DIP) [168] to map proteins to functional terms in the Gene Ontology (GO) [54]. An alternative method by Polacco and Babbitt [122], called Genetic Algorithm Search for Patterns in Structures, or GASPS, constructs short three-dimensional motifs of functional sites consisting of conserved residues through an iterative mutation and selection process. Secondary Structure Matching (SSM) uses a graph-based representation of secondary structure to find similar structural matches to a query structure from the PDB [85]. Laskwoski *et al.* [89] presented a prediction tool based on 3D templates, which are spatial arrangements of three residues representative of functional sites or ligand-binding sites. These can be built from known examples and matched to the query, or the query structure itself can be broken into 'reverse templates' and matched against the PDB.

2.3 Other types of function prediction methods

In addition to sequence and structure, there are tools that incorporate indirect information from scientific literature and association networks. Jaeger *et al.* [73] predicted functions for unannotated proteins using conserved protein interaction networks and supported the predictions using information from literature. Gabow *et al.* [50] showed

that including information about protein co-occurrence in abstracts improved performance of a protein-protein interaction network-based function prediction algorithm.

There are also integrated servers that wrap several or even dozens of methods – sequence-based, structure-based, and others – into one tool. One such server is ProFunc [90], which includes BLAST searches, PROSITE, Pfam, SUPERFAMILY, SSM, and 3D templates, among others. A recent study tested ProFunc's usefulness in predicting function for structural genomics targets and found that the structure-based SSM and 3D templates were most effective [159]. ProFAT is another web-based tool that integrates sequence database search, structural fold recognition, and text mining to predict function for protein sequences [21]. JAFA, like ProFunc, aggregates and reports the results from several other programs [47].

Despite the advances made in protein function prediction and the vast array of available tools, the field still faces many challenges. One is the fact that the number of proteins with unknown function that bear little resemblance in sequence and structure to known proteins is growing rapidly [27]. Function prediction methods that rely on sequence or fold similarity to known proteins will thus be of limited value; indeed, often times the only results returned from these methods are matches to other proteins with unknown function. Another important problem is the difficulty of going from prediction to experimental validation [48]. Function is multifaceted and often cannot be placed neatly into the various classifications we devise. The output of a tool may be as broad as a functional family from Pfam, a match to a particular protein or fold, enzymatic classification such as an EC number [34], or a specific location in the protein structure as from 3D templates or FEATURE (described in Section 2.4). This makes it very difficult to compare predictions and assess the accuracy of predictions. Therefore, it is important to have methods that do not depend on direct sequence

or structure matching and that employ descriptive representations of function for guiding further investigation.

2.4 The FEATURE framework for protein function annotation

This dissertation work builds upon a robust function recognition algorithm called FEATURE [162, 59] which examines 3D environments of molecules in a way that is neither strictly sequence- nor fold-based. The FEATURE system can be broken down into three major components: the way in which sites, or local protein microenvironments, are represented; model building and supervised machine learning methods; and site scoring and model evaluation. FEATURE is flexible in the sense that each of these three components is adaptable to the specific needs of an application.

2.4.1 Microenvironment representation

One of the most important aspects of any structure-based protein function modeling system is how information about a protein is represented and calculated. FEATURE models a local protein microenvironment using a large number of physicochemical properties calculated at varying distances from the site (see Figure 2.1A for a simplified example). A site is defined as a 3D location in a protein structure, and its microenvironment is defined as a sphere centered on that location. In the typical use of FEATURE, 80 physicochemical properties (listed in Table 2.1) are computed in each of six 1.25 Å thick spherical shells – from 0 to 1.25, 1.25 to 2.5, 2.5 to 3.75, etc, up to 7.5 Å. A FEATURE vector thus represents a site as a vector of 480 values (see Figure 2.1B for a simplified example). The FEATURE method has also been

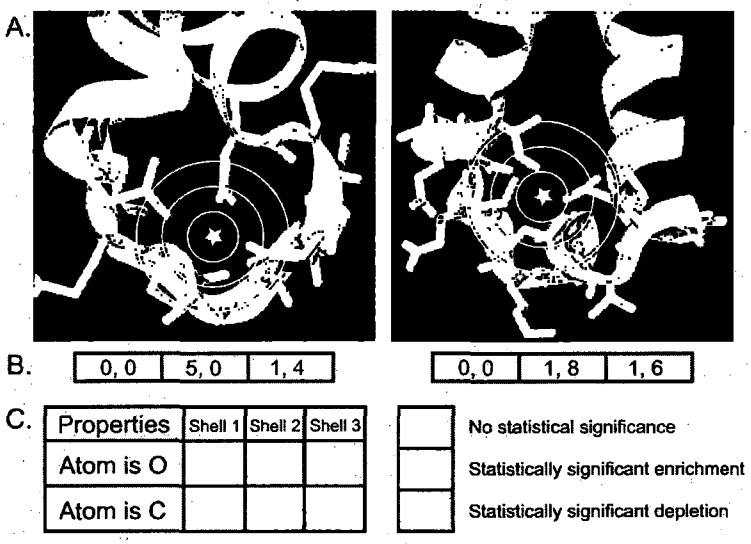


Figure 2.1: Simplified example for building a FEATURE model. **A.** An example positive site (left) and negative site (right), and their respective microenvironments. Properties are calculated in concentric spherical shells centered on each site (star symbol). **B.** FEATURE vectors calculated from A, with oxygen atom count being the first property, and carbon atom count the second. The vectors are divided by shell for clarity. **C.** An example of a visualized FEATURE model, or fingerprint, is shown, based on A and B.

tested successfully on other segmentations of volume, such as a cubic lattice [8, 9]. The concentric spherical shells representation has advantages and disadvantages. One disadvantage is that information about orientation and the relative position of atoms is discarded. Even so, discrete shells are favorable because they allow statistics to be gathered over the relevant volumes and calculation is relatively efficient, which allows FEATURE to serve as an initial filter for more expensive structure-based function prediction methods. Further advantages of this representation include unambiguous definition of a site as a single point in a protein structure, accurate capture of properties of a cumulative nature such as partial charge, and computational efficiency. The use of a single central point for each site means that models can be built with minimal prior knowledge of the geometry of the site – in other words, there is no need to establish other conserved points with which to define a non-spherical coordinate system. Importantly, the use of comprehensible physical and chemical features make the resulting models straightforward to interpret.

Atom-based	Residue-based	Secondary structure-based
ATOM-TYPE-IS-C	RESIDUE-NAME-IS-ALA	SECONDARY-STRUCTURE1-IS-3HELIX
ATOM-TYPE-IS-CT	RESIDUE-NAME-IS-ARG	SECONDARY-STRUCTURE1-IS-4HELIX
ATOM-TYPE-IS-Ca	RESIDUE-NAME-IS-ASN	SECONDARY-STRUCTURE1-IS-4HELIX
ATOM-TYPE-IS-N	RESIDUE-NAME-IS-ASP	SECONDARY-STRUCTURE1-IS-BRIDGE
ATOM-TYPE-IS-N2	RESIDUE-NAME-IS-CYS	SECONDARY-STRUCTURE1-IS-STRAND
ATOM-TYPE-IS-N3	RESIDUE-NAME-IS-GLN	SECONDARY-STRUCTURE1-IS-TURN
ATOM-TYPE-IS-Na	RESIDUE-NAME-IS-GLU	SECONDARY-STRUCTURE1-IS-BEND
ATOM-TYPE-IS-O	RESIDUE-NAME-IS-GLY	SECONDARY-STRUCTURE1-IS-COIL
ATOM-TYPE-IS-O2	RESIDUE-NAME-IS-HIS	SECONDARY-STRUCTURE1-IS-HET
ATOM-TYPE-IS-OH	RESIDUE-NAME-IS-ILE	SECONDARY-STRUCTURE1-IS-UNKNOWN
ATOM-TYPE-IS-S	RESIDUE-NAME-IS-LEU	SECONDARY-STRUCTURE2-IS-HELIX
ATOM-TYPE-IS-SH	RESIDUE-NAME-IS-LYS	SECONDARY-STRUCTURE2-IS-BETA
ATOM-TYPE-IS-OTHER	RESIDUE-NAME-IS-MET	SECONDARY-STRUCTURE2-IS-COIL
ATOM-NAME-IS-ANY	RESIDUE-NAME-IS-PHE	SECONDARY-STRUCTURE2-IS-HET
ATOM-NAME-IS-C	RESIDUE-NAME-IS-PRO	SECONDARY-STRUCTURE2-IS-UNKNOWN
ATOM-NAME-IS-N	RESIDUE-NAME-IS-SER	
ATOM-NAME-IS-O	RESIDUE-NAME-IS-THR	
ATOM-NAME-IS-S	RESIDUE-NAME-IS-TRP	
ATOM-NAME-IS-OTHER	RESIDUE-NAME-IS-TYR	
HYDROXYL	RESIDUE-NAME-IS-VAL	
AMIDE	RESIDUE-NAME-IS-HOH	
AMINE	RESIDUE-NAME-IS-OTHER	
CARBONYL	CLASS1-IS-HYDROPHOBIC	
RING-SYSTEM	CLASS1-IS-CHARGED	
PEPTIDE	CLASS1-IS-POLAR	
	CLASS1-IS-UNKNOWN	
	CLASS2-IS-NONPOLAR	
	CLASS2-IS-POLAR	
	CLASS2-IS-BASIC	
	CLASS2-IS-ACIDIC	
	CLASS2-IS-UNKNOWN	
	PARTIAL-CHARGE	
	VDW-VOLUME	
	CHARGE	
	CHARGE-WITH-HIS	
	NEG-CHARGE	
	POS-CHARGE	
	HYDROPHOBICITY	
	MOBILITY	
	SOLVENT-ACCESSIBILITY	

Table 2.1: Physichochemical properties used by the FEATURE algorithm. FEATURE represents local microenvironments by determining the values of physicochemical properties in each of six concentric, spherical shells centered on the site of interest. Properties include those at the atom level, residue level, and secondary structure level.

2.4.2 Model building by supervised machine learning

FEATURE uses supervised machine learning to combine significant properties into a model that can classify functional sites. To build a model, or description of a functional site, FEATURE requires two training sets: positive sites, which are 3D locations associated with positive examples of the function to be modeled; and negative sites, which are 3D locations not known to be associated with the function. Negative sites can be chosen manually, or randomly sampled from the PDB to have a similar range of atom densities compared to the positive sites. FEATURE vectors are calculated for each site in each training set.

Given a set of FEATURE vectors, a distribution of values is then collected for each property in each shell (Figure 2.1B). We determine whether a property is significantly enriched, significantly depleted, or not significantly different in positive sites compared to negative sites in a given shell using the positive and negative training set distributions. The significance of a property for distinguishing sites from negative sites is calculated over all properties in all shells, and naïve Bayes [40] is used to weight the properties most informative for distinguishing the positive and negative sites. FEATURE models are visualized using "fingerprints", which are color-coded grids that depict the significance of each property in each shell (Figure 2.1C). The choice of negative sites is important as it defines the background distribution and thus determines which features will be considered useful in identifying sites. Different models can result based on different strategies for defining negative sites.

2.4.3 Site scoring and internal model evaluation

In order to determine performance statistics and score cutoffs for classification, the training sets are scored with the model, and sensitivity and specificity are estimated

through k-fold cross-validation. Scores are calculated using a naïve Bayes scoring function, which operates on the assumption that the probability of a site belonging to a particular class is conditioned on the individual probabilities of observed, independent features. In the case of FEATURE, the features correspond to the physicochemical properties calculated in each shell, and their probabilities are derived from the training set distributions. A site's score is then the sum of the probabilities of obtaining an observed feature value given that the site is a positive site, taken over all significant features ν_i in the model:

$$Score = \sum_i \log \left[\frac{P(site|\nu_i)}{P(site)} \right]$$

Score cutoffs are usually based on desired performance, and, as a default, are set to achieve 99% specificity on the training sets, as determined by cross-validation. In k-fold cross-validation, the training set is divided into k groups, and a model is trained on all but one of the groups and tested on the left out group. Once a model is built and score cutoffs defined, potential sites can be scored using that model. FEATURE vectors are calculated for candidate sites in the same way as was done for training sites during model building, and scored using the same naïve Bayes scoring function. The resulting scores indicate the likelihood that the potential site is a positive site, depending on the score cutoff for that model. When possible, the validity of every new model is assessed with an independent test set.

2.4.4 FEATURE in practice

Creating a new model involves a typical workflow (see Figure 2.2) that begins by choosing a function of interest and defining a biologically reasonable definition of

the Cartesian center point for that function (*e.g.* the central position in a binding site or the position of a key atom in an active site). Positive and negative training sets are then created and used to train the model. Cross-validation of the model on the training sets allows definition of score cutoffs based on desired performance, and whenever an independent test set is available, model performance can be further assessed. Once a model is built and a score cutoff has been defined, FEATURE can predict functional sites in structures of interest.

An especially important step in model training is the selection of sites for the positive training set, and, in order to tune performance, the negative training set. The training sets for the first FEATURE models were built and verified by hand using published literature; these include the calcium-binding [161] and ATP-binding [162] site models. The calcium-binding model has especially good performance, and is currently being used in multiple ongoing projects to expand FEATURE's capabilities and applicability. The recently published zinc-binding model [43], which involved a mixture of manual and automated approaches, is the best performing zinc-binding predictor currently available. We have also applied FEATURE to function prediction in RNA

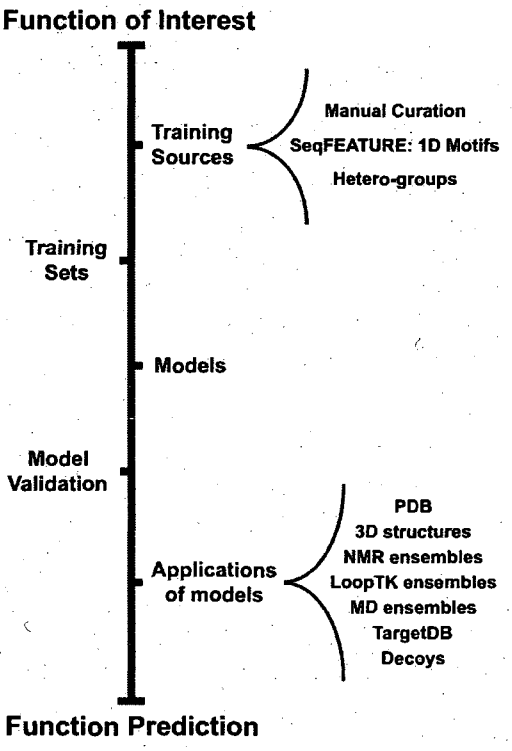


Figure 2.2: The FEATURE framework. To build a FEATURE model, one must first define the function of interest and create positive and negative training sets from the appropriate data sources. Then, the model is trained and evaluated on the training sets. The validated model can be used for function prediction in a variety of ways.

structures with two magnesium-binding models, one for diffuse binding and one for site-specific binding [10].

From its manually-curated beginnings, FEATURE has expanded to include automatic generation of training sets using sequence motifs [95, 165], PDB annotations, and even a clustering of FEATURE vectors encompassing a non-redundant subset of the entire PDB [169]. In addition, FEATURE can be applied to many problems in structural biology such as modeling dynamics of functional sites [53] and decoy and loop filtering for structure prediction.

2.5 Protein function resources

Many of the tools already mentioned are coupled with databases describing the functional motifs or protein families, such as Pfam and InterPro. In addition, there are compendia of specific types of functional sites or annotations, specialized databases for particular organisms or types of proteins, and comprehensive knowledgebases. Uniprot is the largest protein knowledgebase, containing information from primary literature, annotations and predictions from other databases, and free text comments [155]. The manually-reviewed and unreviewed portions are known as Swiss-Prot [17] and TrEMBL, respectively. Organism-specific information is available in the Human Protein Reference Database [125] and FlyBase [164], among others. Databases containing biological pathway information include KEGG [80] and BioCyc [26], and protein-protein interaction data can be found in DIP and STRING [76]. The Catalytic Site Atlas contains descriptions of enzyme catalytic sites [124], and Pegg *et al.* [118, 119] have created a database of enzyme structure-function linkages. PDBsum contains structural and functional analyses and predictions for PDB structures [88].

Chapter 3

The SeqFEATURE library of 3D functional site models

Although many sequence-based function prediction methods exist, the rapidly increasing number of novel protein structures containing very little sequence similarity to known proteins creates a need for methods that incorporate other types of information. Structure-based methods are available, but most rely on structural similarity, which SG structures also tend to lack. FEATURE (see Section 2.4) is especially suited for this problem since it is structure-based, but not dependent on exact matches; however, building functional site models requires identification of positive and negative training examples. Previously, this was a manual and often time-consuming process.

To address this, we developed a method called SeqFEATURE which automatically selects training sets using 1D sequence motifs. The training sets are then used to build functional site models which can be used to scan protein structures for function

The work presented in this chapter builds upon work by Mike P. Liang [95]. Portions of this chapter also appeared in the following paper: Wu S, Liang MP, Altman RB. (2008) The SeqFEATURE library of 3D functional site models: comparison to existing methods and applications to protein function annotation. Genome Biology 9:R8.

(see Figure 3.1). This approach was first conceptualized by Mike Liang [95], and is expanded, implemented, and validated here. Using this approach, we built a library of 3D functional site models, cross-validated and characterized their performance, and then compared their performance to a suite of state-of-the-art function prediction tools, both sequence- and structure-based. We show that SeqFEATURE models produce fewer false positive and false negative predictions than their 1D counterparts, are generally competitive with other methods, and, most importantly, are more robust than other methods when sequence identity and structural similarity are low. We have also scanned the entire PDB with the library, including SG structures with unknown function, resulting in interesting predictions.

This chapter describes the methodology and results for the creation of the SeqFEATURE library, including its validation, comparison, and application.

3.1 Methods

3.1.1 Training set selection

SeqFEATURE adds to the FEATURE framework by using one-dimensional sequence motifs as seeds for generating training sets of structural examples. This method was first introduced in a single application to calcium binding by EF-hand motifs [95], and is extended and applied here into a full library of functional site models.

To build the library of models, we extracted structural examples of PROSITE functional site patterns from the ASTRAL40 compendium [22], which is a nonredundant subset of protein domains in the PDB. PROSITE patterns are regular expressions that specify the amino acids permitted at each position of the motif. We defined functional site centers to be the functional atom(s) of annotated functional residues

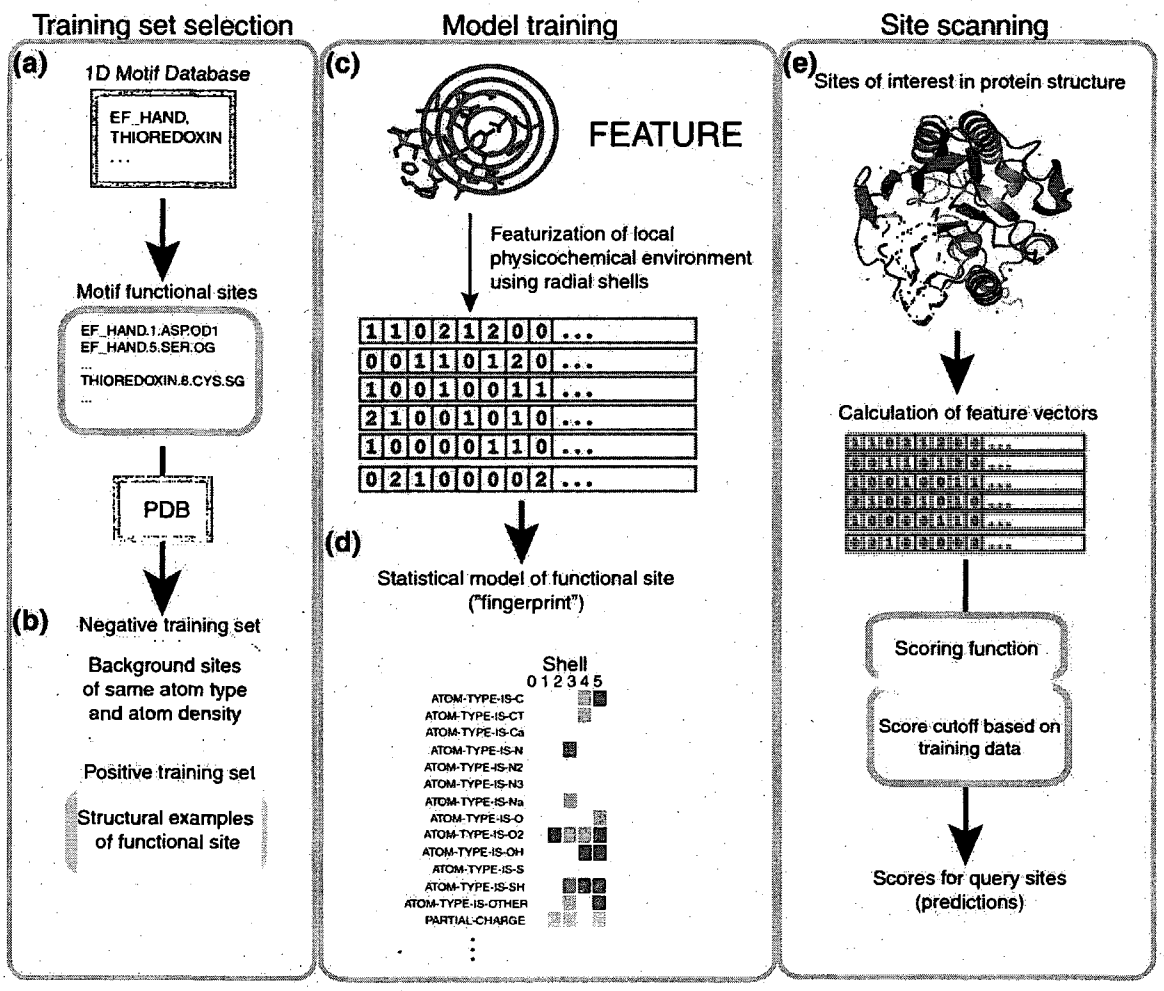


Figure 3.1: Overview of the SeqFEATURE pipeline. SeqFEATURE forms training sets by (a) extracting sequence motifs from PROSITE and identifying the annotated functional amino acids. (b) We extract examples of the sequence motif with known three-dimensional structure in the PDB and center FEATURE training sites on each functional atom of each functional amino acid annotated in the PROSITE pattern. We choose negative sites matched for atom density randomly from the PDB that do not contain the function. (c) FEATURE then creates a model of the sites by summarizing the chemical and physical features found in concentric shells around the functional atom center. (d) The resulting fingerprint specifies the properties that are in relative abundance or paucity in the site, representing the model. (e) Sites in a protein of interest are converted into feature vectors and scored with the model using a naïve Bayes scoring function, and predictions are made using score cutoffs, which can be based on desired performance statistics. The scores are calibrated into Z-scores using the training set used to derive each model.

in each pattern; e.g. the gamma oxygen of serine, or SER.OG. For patterns with multiple functional residues or multiple functional atoms, we built multiple models for the same PROSITE pattern. For example, the PROSITE pattern EGF_1 has functional cysteine residues at positions 1, 3, and 7, so there are three models centered on the three functional atoms in this pattern - EGF_1.1.CYS.SG, EGF_1.3.CYS.SG, and EGF_1.7.CYS.SG. Models derived from PROSITE are always named using a four-part naming scheme specifying the motif, the position in the motif, the residue at that position, and the atom within that residue upon which the model is centered. See Table 3.1.1 for a complete list of SeqFEATURE models.

Table 3.1: SeqFEATURE models built from PROSITE motifs.

PROSITE Pattern	Positions	Residue	Atom(s)
2FE2S_FERREDOXIN	1, 6, 9	CYS	SG
4FE4S_FERREDOXIN	1, 3, 5, 7	CYS	SG
AA_TRANSFER_CLASS_1	4	LYS	NZ
AA_TRANSFER_CLASS_2	4	LYS	NZ
AA_TRANSFER_CLASS_3	19	LYS	NZ
ADH_SHORT	3	TYR	OH
ADH_ZINC	2	HIS	ND1, NE2
ADX	6, 9	CYS	SG
ALDEHYDE_DEHYDR_CYS	6	CYS	SG
ALDEHYDE_DEHYDR_GLU	2	GLU	OE1, OE2
ASP_PROTEASE	4	ASP	OD1, OD2
ASX_HYDROXYL	3	ASN	ND2, OD1
BETA_LACTAMASE_A	5	SER	OG
BETA_LACTAMASE_B_1	4, 6	HIS	ND1, NE2
	8	ASP	OD1, OD2
BPTI_KUNITZ_1	4, 8	CYS	SG
C_TYPE_LECTIN_1	1	CYS	SG
CARBOXYLESTERASE_B_1	11	SER	OG
CARBOXYLESTERASE_B_2	3	CYS	SG
CHITINASE_18	9	GLU	OE1, OE2
COPPER_BLUE	7	CYS	SG
	11	HIS	ND1, NE2
CYTOCHROME_P450	8	CYS	SG
EF_HAND	1, 3, 5, 9	ASP	OD1, OD2
	3, 5, 9	ASN	ND2, OD1
	5, 9	SER	OG
	7, 9	THR	OG1
	7, 12	TYR	OH

	7	GLU	OE1, OE2
	7	LYS	NZ
EGF_1	1, 3, 7	CYS	SG
EGF_2	1, 3, 8	CYS	SG
GLYCOSYL_HYDROL_F5	7	GLU	OE1, OE2
GLYCOSYL_HYDROL_F10	7	GLU	OE1, OE2
HIPIP	1, 7	CYS	SG
HMA_1	5, 8	CYS	SG
IG_MHC	3	CYS	SG
IMP_1	4	ASP	OD1, OD2
KAZAL	1, 3, 7, 9	CYS	SG
LIPASE_SER	7	SER	OG
LIPOYL	9	LYS	NZ
PA2_HIS	4	HIS	ND1, NE2
PEROXIDASE_1	8	HIS	ND1, NE2
PEROXIDASE_2	8	HIS	ND1, NE2
PHOSPHOPANTETHEINE	6	SER	OG
PROTEIN_KINASE_ST	5	ASP	OD1, OD2
PTS_HPR_SER	5	SER	OG
RNASE_T2_1	4	HIS	ND1, NE2
SHIGA_RICIN	5	GLU	OE1, OE2
	8	ARG	NE, NH1, NH2
SMALL_CYTOKINES_CC	1, 2, 11, 17	CYS	SG
SNAKE_TOXIN	2, 4, 7, 8	CYS	SG
SUBTILASE_ASP	5	ASP	OD1, OD2
THIOL_PROTEASE_ASN	6	ASN	ND2, OD1
THIOL_PROTEASE_HIS	3	HIS	ND1, NE2
THIOREDOXIN	8, 11	CYS	SG
TRYPSIN_HIS	5	HIS	ND1, NE2
TRYPSIN_SER	6	SER	OG
TYR_PHOSPHATASE_1	3	CYS	SG
UBIQUITIN_CONJUGAT_1	10	CYS	SG
ZINC_FINGER_C2H2_1	1, 3	CYS	SG
	7, 9	HIS	ND1, NE2
ZINC_PROTEASE	3, 7	HIS	ND1, NE2
	4	GLU	OE1, OE2

Positive training sets consist of PDB coordinates of functional atoms as described above, extracted from structures containing that particular pattern (see Appendix A.2). We required training sets to have a minimum of five structural examples. We selected negative training sets randomly from identical residues in the rest of the PDB whose atom compositions and densities are similar to the positive sites. In order to define the background distribution of the functional site environments, we

used a thousand times as many negative sites as positive sites for each model, when possible, but never less than 4,000.

3.1.2 Model cross-validation and evaluation

We internally evaluated each model using five-fold cross validation by partitioning the positive and negative training sets randomly into five blocks. For each run, we used four blocks to build the model and tested performance on the remaining block. To compare results across runs, we transformed the scores into Z-scores by standardizing to the mean and standard deviation of the negative score distribution.

To measure performance, we use receiver operating characteristic (ROC) curves, which plot the true positive rate (sensitivity, or the ratio of true positive predictions to all true positives) against the false positive rate (1-specificity, or the ratio of false positive predictions to all true positives) at varying Z-score cutoffs. We also plot positive predictive value (PPV) against sensitivity to gauge the performance of a model. Sensitivity, specificity, and PPV are calculated as follows:

$$\text{Sensitivity} = \frac{\#\text{ of true positive predictions}}{\text{total }\#\text{ of true positives}}$$

$$\text{Specificity} = \frac{\#\text{ of true negative predictions}}{\text{total }\#\text{ of true negatives}}$$

$$\text{Positive predictive value} = \frac{\#\text{ of true positive predictions}}{\text{total }\#\text{ of positive predictions}}$$

The AUC estimates the probability that a random positive site will be scored higher than a random negative site, and provides a summary measure of the performance of the model. The final models used all of the training examples, and include score cutoffs calculated for 95%, 99%, and 100% specificity based on cross-validation.

3.1.3 Procedure for comparison to other methods

The manually curated PROSITE record for each pattern contains known true positives, false positives, and false negatives predicted by that pattern, listed using Swiss-Prot identifiers. We treated each Swiss-Prot ID as a unique protein. Using existing mappings between Swiss-Prot and the PDB, we converted each list into a list of corresponding PDB structures to use as input to SeqFEATURE and other structure-based methods. Thus, our positive test set consisted of Swiss-Prot IDs and PDB structures for proteins annotated as true positives and false negatives in PROSITE, and our negative test set consisted of Swiss-Prot IDs and PDB structures for proteins annotated as false positives. We removed all positive training set structures from the test sets and filtered the test structures to ensure that they contained the functional regions described by the relevant PROSITE pattern.

Defining function for evaluation purposes

Using these test sets, we compared performance between PROSITE, Pfam, Gene3D, HMMPanther, SSM, 3D templates (reverse template type), and SeqFEATURE (see Chapter 2 for a description of these other methods). In order to ensure consistency across the comparisons, we restricted the analysis to PROSITE patterns that had at least one SeqFEATURE model with an AUC >0.75 and that also mapped unambiguously to classifications used by the tool being compared, using publicly available mappings.

Unambiguous assignments were those for which either 100% of the training set mapped to the same Pfam family, or for which the Pfam family clearly matched the PROSITE pattern (for example, PROSITE pattern GLYCOSYL_HYDROL_F10 and Pfam family 'Glyco_hydro_10'). Forty-two PROSITE motifs had both an AUC

>0.75 and a positive test set independent of the training set (TRYPSIN_HIS was excluded due to it being nearly identical to TRYPSIN_SER), and, of these, 31 mapped unambiguously to Pfam, 12 to Panther, and 29 to Gene3D.

Comparing sequence-based methods to SeqFEATURE

Because structure-based methods such as 3D templates and SSM are more computationally expensive to run than SeqFEATURE and the sequence-based methods, we split the comparison into two parts. The first part compared the sequence-based methods – PROSITE, Pfam, HMMPanther, and Gene3D – to SeqFEATURE, and covered the unambiguous portions of the test sets in their entirety. PROSITE's predictions came directly from its annotations. For the other sequence-based methods, we analyzed the test set proteins using each tool and marked a protein as a positive prediction if at least one of its predictions matched the unambiguous assignment for the pattern being tested. HMMPanther and Gene3D were run from the InterPro servers using the stand-alone downloadable Perl client [127]. Pfam's predictions were taken directly from their publicly available mapping file. For SeqFEATURE, we classified a protein as positive if at least one of its mapped PDB structures scored above the specified cutoff for at least one model derived from that pattern. Since SeqFEATURE cutoffs are variable, we tested performance at 95%, 99%, and 100% specificity cutoffs.

Comparing structure-based methods to SeqFEATURE

To compare SSM, 3D templates, and SeqFEATURE, we limited our test sites to those derived from PROSITE patterns that mapped to EC numbers. Since 3D templates (reverse template type) and SSM both return protein structures rather than a named

function as output, we used EC numbers to evaluate predictions made by SSM and 3D templates. We determined the set of EC numbers corresponding to each pattern's training set and randomly sampled 29 positive sites and 15 negative sites from the EC-compatible subset of test sites. We then took the top prediction below 95% sequence identity to the query for each test site from SSM and 3D templates that had an EC number, and considered it a positive prediction if the EC number matched any of the EC numbers assigned to the relevant PROSITE pattern. We determined SeqFEATURE predictions by evaluating whether each structure scored above the 95%, 99%, and 100% cutoffs for at least one model derived from the appropriate pattern.

3.1.4 Evaluating performance at low sequence identity and structural similarity

We also compared the sequence-based methods to SeqFEATURE using low sequence identity test sets. We computed all pairwise sequence alignments between structures in the positive test set and the training set for each pattern using JAligner, a freely available Smith-Waterman alignment software package [74], and constructed a new test set consisting of those test structures that had less than 35% sequence identity to structures in their corresponding training set. We broke down the test set according to sequence identity thresholds differing by 2% (<35%, <33%, and so on, down to <25%). To prevent any pattern from dominating the test set, we further filtered the test set so that no pattern had more than one site in each sequence identity range by selecting one at random when multiple sites were present. We then looked up the predictions from the sequence-based methods for the low sequence identity test set at each of these thresholds.

From the low sequence identity test set, we conducted pairwise structural similarity searches between each structure and the structures in the corresponding training sets using DALI, a freely-available tool for calculating structural similarity [36]. We discarded any structure that matched a training set structure with a DALI Z-score greater than 10.0. The remaining structures all had no significant matches, or only low-confidence matches, to their positive training sets. We then looked up the predictions from 3D templates, SSM, and SeqFEATURE (at the three different cutoffs) for the low structural similarity test set.

3.1.5 Protein Data Bank scan procedure

Any PDB structure can be scanned with any SeqFEATURE model to generate a list of predictions. We conducted a full scan of the March 2006 version of the PDB, which contained about 35,600 structures, about 95% of which were proteins. We extracted lists of each of the relevant potential functional atoms from each protein structure (ARG.NE, ASP.OD1, ASP.OD2, CYS.SG, and so on), including all chains. This resulted in 90,919,770 potential sites. We then scored all of these sites with the corresponding models that were built on that particular type of functional atom. The entire scan (extracting and scoring) took about one day to complete on fourteen parallel processors. To analyze the scan data, we filtered out redundant scores from proteins with multiple, identical chains.

3.1.6 TargetDB prediction analysis

We focused our scan analysis on structures listed in TargetDB, the database for targets from structural genomics centers [29]. Using the headers of released PDB files, we filtered for those that lacked functional annotation; for example, 'STRUCTURAL

GENOMICS,' 'UNKNOWN FUNCTION', 'HYPOTHETICAL PROTEIN', and so on. We scanned these structures with the entire library of SeqFEATURE models and examined the predictions for those hits that satisfied the following two conditions:

1. The prediction was for a model that has an AUC >0.85; and
2. The hit scored above the 100% specificity cutoff or well within the positive Z-score distribution for that model.

We then compared each prediction to the results of PROSITE, Pfam, HMMPanther, Gene3D, SSM, and 3D template searches on those structures, and prioritized cases where the sequence-based methods produced no significant predictions.

3.2 Results

3.2.1 The SeqFEATURE model library

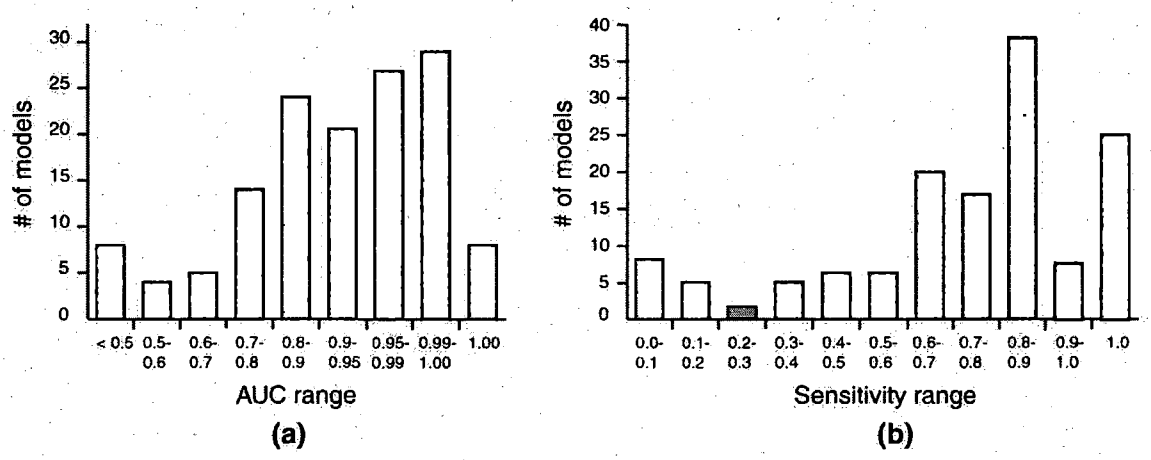


Figure 3.2: Distribution of AUC and sensitivity for all SeqFEATURE models. The majority of models have AUC greater than 0.8 and sensitivity greater than 0.75 in internal cross-validation.

The SeqFEATURE library consists of 136 models derived from 53 PROSITE patterns (Table 3.1.1). Of these models, 105 (77%) have an AUC greater than 0.8, and 64 (47%) have an area under the curve (AUC) greater than 0.95 (Figure 3.2a). Sensitivity at the default 99% specificity cutoff is slightly more variable, but 82% of the models have sensitivity greater than 0.5 and 59% have sensitivity greater than 0.75 (Figure 3.2b).

ROC curves from cross-validation and Z-score distributions of the final models can be used together to evaluate the ability of the model to distinguish true sites from the background. We visualize the separation between the positive and negative sites by plotting the distributions of Z-scores for the positive and negative training examples. Plots of PPV versus sensitivity, also known as precision-recall curves, give the proportion of total hits to the models that are true positives as a function of sensitivity. Representative examples of ROC curves, precision-recall curves, and Z-score distributions for a range of model performances are shown in Figure 3.3.

The sensitivity of the top-performing models (ranked by AUC) is very high in general, especially at the default 99% specificity Z-score cutoffs. Even at 100% specificity a significant proportion of models have greater than 0.75 sensitivity. A wide range of PROSITE patterns is also represented in the top-ranked models, indicating that the method performs well for many different types of functional sites. See Appendix A.1 for a full list of model performance statistics.

3.2.2 Performance compared to other methods

In order to get a more realistic estimate of the library's performance, we constructed a specialized test set from the PROSITE records for each pattern, which contain manually curated annotations of true positives, false positives, and false negatives.

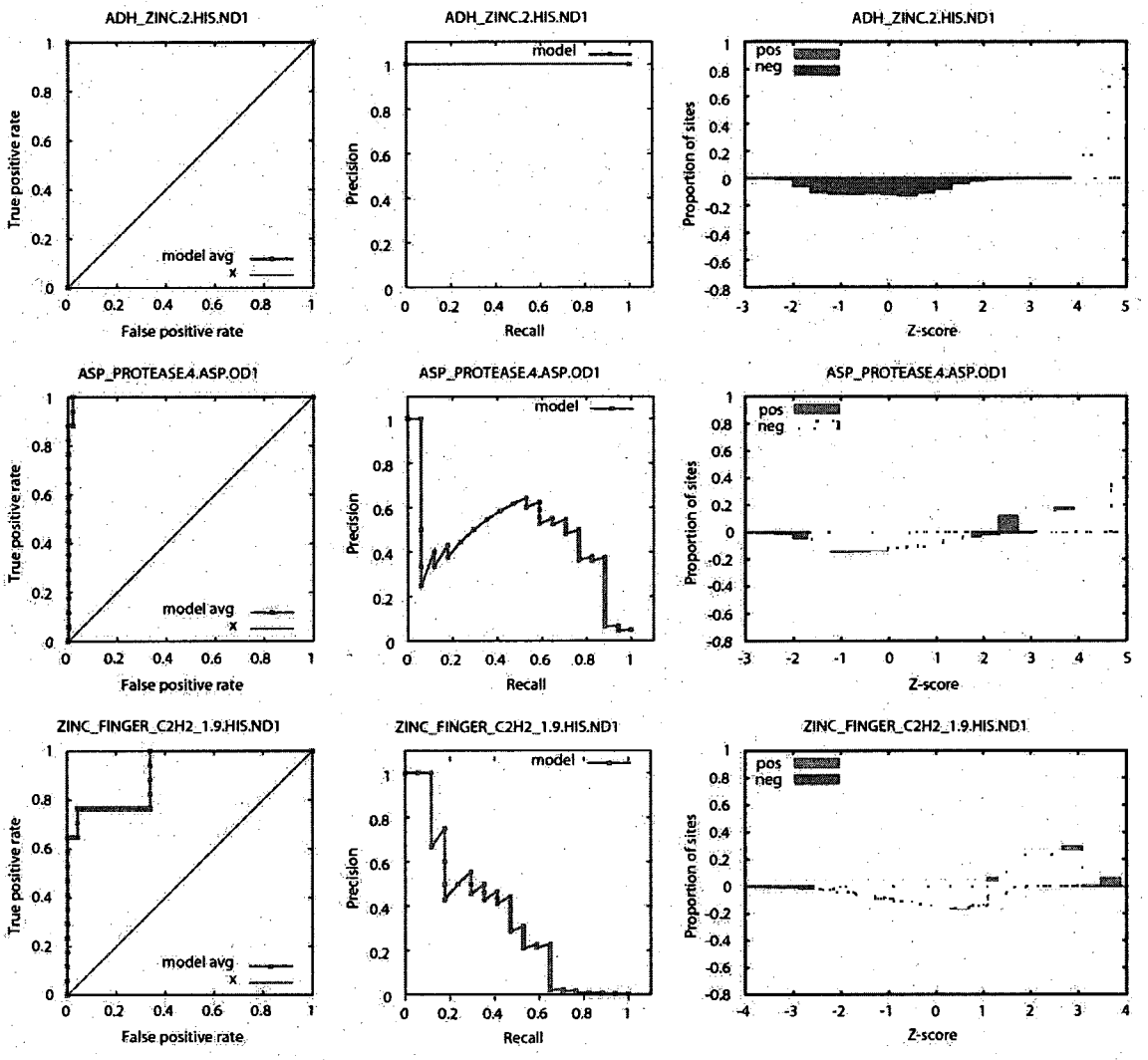


Figure 3.3: Example performance plots for SeqFEATURE models. Plots for three models are shown as representatives of excellent, good, and satisfactory performance. On the left are ROC curves, with blue lines indicating the performance of a random classifier. The middle plots show precision-recall curves, and the rightmost plots show the distribution of normalized Z-scores for positive sites (red) and negative sites (blue) used in training.

The test sets consisted, therefore, of structures that the associated PROSITE pattern is known to detect correctly, falsely predict, and altogether miss.

Importantly, we could directly compare if and where SeqFEATURE outperforms the originating PROSITE pattern. Figure 3.4 shows the numbers of true positives,

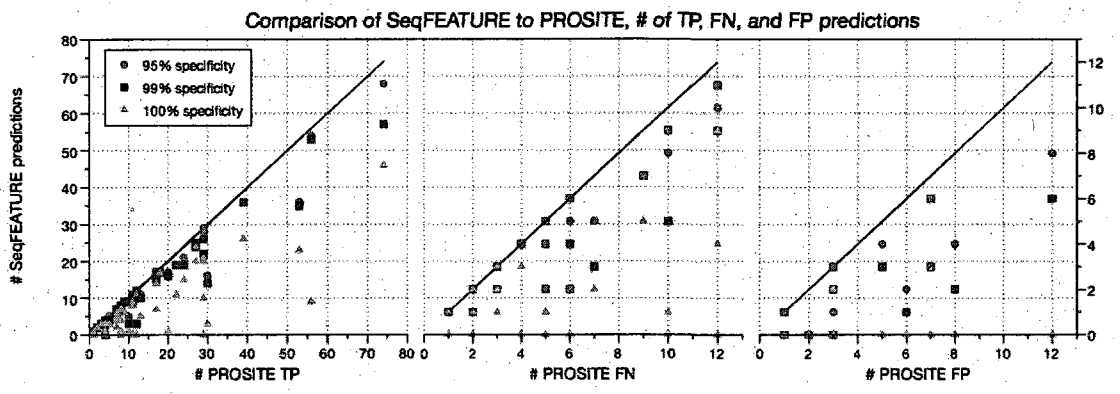


Figure 3.4: Performance on PROSITE-derived test set. We show the number of true positive (TP), false negative (FN), and false positive (FP) predictions for SeqFEATURE at three different specificity-based score cutoffs compared to PROSITE on test sites derived from the corresponding PROSITE patterns. Each dot represents a comparison for one PROSITE pattern, and the PROSITE values represent the maximum possible for each category (solid line). Note that not all patterns had a false negative or false positive test set. In addition, most of SeqFEATURE's incorrect predictions at 95% and 99% specificity cutoffs arise from poor performance on a small subset of patterns.

false negatives, and false positives predicted by SeqFEATURE at varying specificity-based score cutoffs compared to the corresponding PROSITE pattern. Figure 3.5 shows overall numbers of predictions in each category. Since the test sets were derived from PROSITE, the PROSITE values represent the maximum that could possibly be obtained for each type of prediction. The three different cutoffs show tradeoffs in the numbers of true positive, false positive, and false negative predictions made by SeqFEATURE; one can therefore adjust the cutoff to fit desired performance.

When we compared performance between SeqFEATURE, Pfam, HMMPanther, and Gene3D, we found Gene3D to be the best performing method by far, with sensitivity just over 98%, specificity at 85.4%, and PPV at 99% (Table 3.2). Pfam was the second most sensitive method at 93.7%; since it predicted all negative examples (PROSITE false positives) correctly, Pfam had a PPV of 100%. HMMPanther scored slightly below Pfam on its limited test set with a sensitivity of 91.9%; there were not enough examples to evaluate specificity. SeqFEATURE had a sensitivity of 86.2% at

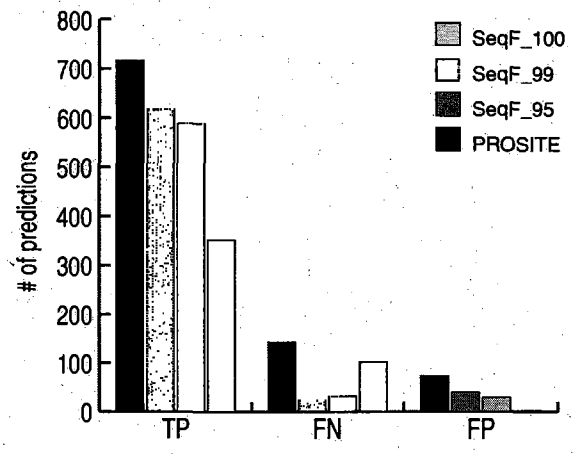


Figure 3.5: Summary performance on PROSITE-derived test set This plot summarizes total numbers of predicted true positives, false negatives, and false positives for PROSITE and SeqFEATURE at 100%, 99%, and 95% specificity cutoffs. While SeqFEATURE does not predict all true positives correctly, it predicts 82% of true positives correctly at the default 99% specificity cutoff. At the same cutoff, SeqFEATURE also predicts about 78% fewer false negatives than PROSITE, and about 60% fewer false positives.

our most lenient cutoff, and specificity and PPV comparable to Pfam and Gene3D at our more stringent cutoffs. Interestingly, all of the sequence-based methods show a marked decrease in sensitivity when evaluated only on positive examples that did not contain the PROSITE motif (that is, PROSITE false negatives). SeqFEATURE, on the other hand, is not as significantly affected by whether the test proteins contain the canonical sequence motifs.

On the randomized sample test set, we were able to compare SeqFEATURE to 3D templates and SSM (Table 3.3). Here, SeqFEATURE's best sensitivity increased to 93%, though its best specificity dropped to 93%. PPV decreased slightly to 94%

	Gene3D	Pfam	HMMPanther	SeqF_95	SeqF_99	SeqF_100
TP sensitivity	**0.998**	**0.937**	0.919	0.862	0.821	0.492
FN sensitivity	**0.907**	0.704	0.532	**0.831**	0.775	0.282
Overall sensitivity	**0.983**	**0.898**	0.831	0.857	0.814	0.457
(FP) Specificity	0.854	**1.000**	—	0.452	0.603	**0.973**
Pos pred value	0.990	**1.000**	—	0.948	0.960	**0.995**

Table 3.2: Comparison of SeqFEATURE to Gene3D, Pfam, and HMMPanther. We evaluated SeqFEATURE at three different specificity-based score cutoffs. SeqFEATURE is competitive but the best sequence-based methods, particularly Gene3D and Pfam, clearly outperform in general. Interestingly, SeqFEATURE performs relatively better on harder cases (false negative sensitivity and false positive specificity). The best two values are bolded in each row.

at the most stringent cutoff. 3D templates performed most well out of the structure-based methods, with 90% sensitivity, 100% specificity, and a PPV of 100%. SSM performed similarly to SeqFEATURE.

	3D templates	SSM	SeqF_95	SeqF_99	SeqF_100
Sensitivity	**0.897**	0.724	**0.931**	0.862	0.552
Specificity	**1.000**	**0.933**	0.600	0.667	**0.933**
Pos pred value	**1.000**	**0.955**	0.818	0.833	0.941
LSS-sensitivity	0.200	0.267	**0.533**	**0.467**	0.133

Table 3.3: Comparison of SeqFEATURE to 3D templates and SSM. SeqFEATURE (at three specificity-based score cutoffs) is again competitive but the other structure-based methods tend to perform better in general. The real gain arises when structural similarity of the target to known proteins is reduced; SeqFEATURE provides robust performance while the other methods perform much less well. The best two values are bolded in each row.

3.2.3 Performance at low sequence identity and structural similarity

Since the goal of many function prediction methods, including SeqFEATURE, is to aid in annotation of solved structural genomics targets, we also compared SeqFEATURE to the sequence-based methods using low sequence identity test sets to mimic the situation in which a newly solved structure has low sequence identity to proteins of known function. As shown in Figure 3.6, the range of error increases as we reduce the sequence identity, making it difficult to derive any definitive conclusions, but the sequence-based methods perform slightly less well overall, particularly on sequences filtered at 30% and 25% identity. Of the sequence-based methods, Gene3D maintains the most consistent performance. In contrast, SeqFEATURE's performance is much more robust to decreases in sequence identity. Note that the test sets were additionally filtered to reduce bias from over-represented PROSITE patterns; some patterns on

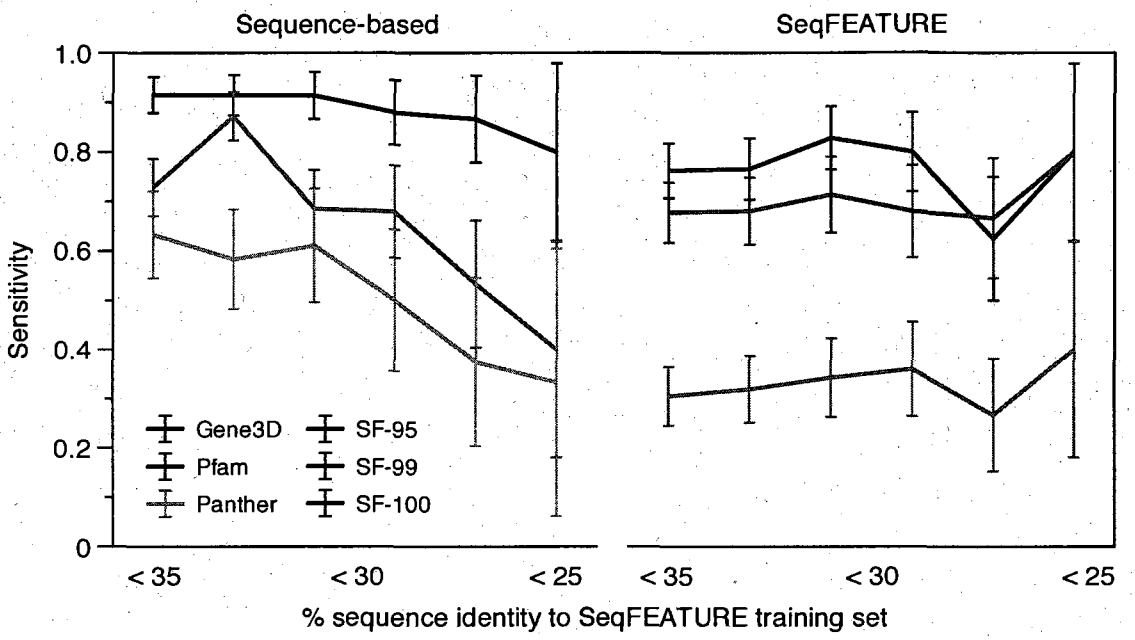


Figure 3.6: Sensitivity trends of SeqFEATURE and sequence-based methods at low sequence identities. We compared the sensitivity of SeqFEATURE at three specificity cutoffs against the sensitivity of Gene3D, Pfam, and HMMPanther on test sets filtered for low sequence identity with respect to the SeqFEATURE training sets. As sequence identity decreases, the sequence-based methods show a trend towards lower sensitivity. In contrast, SeqFEATURE at all three cutoffs shows no such downward trend, indicating robust detection of function even when sequence identity is very low.

which SeqFEATURE performs very well dominated in the lower identity ranges and were filtered out.

To determine whether the degree of structural similarity affects how well different methods predict function, we also constructed a low structural similarity test set using DALI. Although relatively small (15 examples), the low structural similarity test set allows us to approximate the situation of function prediction on novel folds. As shown in Table 3.3, SeqFEATURE performs better at the 95% and 99% specificity cutoffs than the other structure-based methods; its low structural similarity (LSS)-sensitivity is 53% and 47%, respectively, while the LSS-sensitivity values for SSM and 3D templates are both less than 30%.

As observed in a recent study of function prediction for structural genomics targets, it is rare for a single method to outperform others in all cases; likewise, most methods will have their particular niche in which they tend to show the better performance. For this reason, we examined the incorrect predictions to see if there were differences between the methods we compared. Tables 3.4 and 3.5 present the fraction of false predictions in each test category from each of the sequence and structure-based function prediction methods that were correctly classified by SeqFEATURE at the default 0.99 specificity cutoff. In general, SeqFEATURE makes the correct prediction 65% of the time when the sequence-based methods make false predictions, with especially marked improvement for certain categories over certain methods, such as false negative predictions made by Gene3D.

A similar improvement is seen when we examine the low sequence identity set alone, but the greatest improvement occurs in the lowest sequence identity category, with 96% of false predictions correctly classified. Recovery of false predictions made by structure-based methods is not as substantial but still significant, with 45% of missed positives correctly classified by SeqFEATURE. No improvement over this recovery rate is observed in the low structural similarity test set.

3.2.4 Predictions of function for structural genomics targets

As of November 2007, TargetDB contained about 5,250 targets with structures released in the PDB; of these, about 1,500 were labeled only with 'structural genomics', 'unknown function', or 'hypothetical protein' in the PDB file header. Using the criteria described in Section 3.1.6, we found 35 potential functional sites. We added one more predicted functional site that did not quite satisfy the criteria but had several such hits for multiple models for the same function, resulting in a total of 36

	Gene 3D	Pfam	HMMPanther	Total
TP	1/1 (100%)	23/35 (66%)	2/3 (67%)	26/39 (67%)
FN	9/10 (90%)	24/34 (71%)	13/22 (59%)	46/66 (70%)
FP	0/5 (0%)	—	—	0/5 (0%)
				72/110 (65%)
30-35% ID	2/2 (100%)	816 (50%)	5/13 (38%)	15/31 (48%)
25-30% ID	1/2 (50%)	4/8 (50%)	2/4 (50%)	7/14 (50%)
20-25% ID	6/6 (100%)	8/8 (100%)	8/9 (89%)	22/23 (96%)
				44/68 (65%)

Table 3.4: Recovery of false predictions from sequence-based methods by SeqFEATURE. For each method, we show the fraction of false predictions made by each method in each category that SeqFEATURE (at the default 99% specificity cutoff) classifies correctly. Note that SeqFEATURE has an especially good recovery rate of 96% in the lowest sequence identity range.

	SSM	3D templates	Total
Positive set	4/8 (50%)	1/3 (33%)	5/11 (45%)
Negative set	0/1 (0%)	0/3 (0%)	0/4 (0%)
Low-SS	4/11 (36%)	5/12 (41%)	9/23 (39%)

Table 3.5: Recovery of false predictions from structure-based methods by SeqFEATURE. Although the recovery rate is not as high as for the sequence-based methods, SeqFEATURE is able to classify correctly about 40% of the incorrect predictions made by SSM and 3D templates.

high-confidence predictions. Since publication of the SeqFEATURE work, we have updated our TargetDB predictions with 191 structures with unknown function added after November 2007, resulting in 12 additional predictions (see Appendix B for all predictions). We compare our predictions to those of PROSITE, Pfam, Gene3D, HMMPanther, SSM and 3D templates for the same structures.

In examining these structures, we found that some of them, though labeled as 'unknown function', actually had some functional annotation and, thus, we could determine the plausibility of our prediction. For example, PDB structure 1XRI is described

as a putative phosphatase, and had a high scoring hit for the TYR_PHOSPHATASE-_1.3.CYS.SG model. All of the other methods also detected phosphatase activity. Another example is 2E72, described as a zinc-finger containing protein, which hit our ZINC_FINGER_C2H2_1.1.CYS.SG model and for which Pfam, Gene3D, HMMPanther, SSM, and 3D templates all predicted zinc finger motifs. More interesting, however, are predictions for structures that fail to garner any high-confidence predictions from PROSITE, Pfam, Gene3D, or HMMPanther. Table 3.6 presents four intriguing cases. In all of these cases, only SeqFEATURE gives a high-confidence prediction, though 3D templates and SSM sometimes offer matches to putative functions or have low-confidence predictions. In contrast, the SeqFEATURE predictions have relatively high Z-scores compared to the training set distributions.

PDBID	SeqFEATURE model	Site	Z-score	Other predictions
3BJQ	ZINC_PROTEASE.4.GLU.OE1	GLU96:A	3.774	SSM: bacteriophage prohead II; 3D templates: zinc-finger C2H2
2EJQ	ZINC_PROTEASE.4.GLU.OE1	GLU123:F	4.574	3D templates: Probable anthrax toxin lethal factor
2OGF	EF_HAND.9.THR.OG1	THR17:D	4.675	SSM: Aminopeptidase (Z-score = 2.7)
2OX6	EF_HAND.9.ASN.OD1	ASN8:B	4.102	3D templates: Probable Zn enzyme

Table 3.6: Predicted functions for TargetDB structures with unknown function. The SeqFEATURE model that produced the high scoring hit is shown along with the location of the predicted site and the Z-score that the site received from the model. The best predictions from the other sequence and structure-based methods, when a prediction was available, are also shown for comparison. No sequence-based methods produced any significant predictions.

We selected one of these predictions for further investigation (see Figure 3.7). 3BJQ is a phage-related protein isolated from Bordatella bronchiseptica, a species of pathogenic bacteria. There is a page devoted to this protein [150] on The Open Protein Structure Annotation Network (TOPSAN) website [151], where it is noted

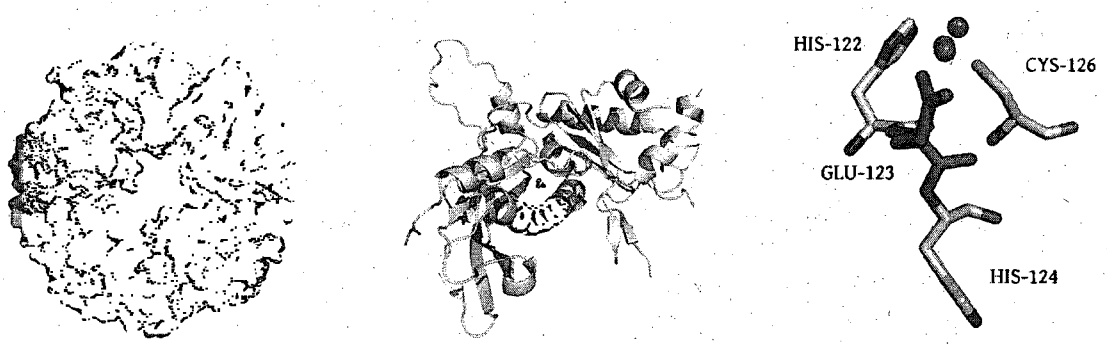


Figure 3.7: Analysis of 3BJQ. 3BJQ forms a decameric channel-like structure in solution through stacking of two pentamers (a). In (b), one of the subunits is shown with the predicted zinc protease residue highlighted in red. Zinc binding sites (grey spheres) are predicted in each subunit as well, in close proximity to the predicted zinc protease active site residue. In (c), we show the predicted site – a glutamate residue (in red) – flanked by two histidine residues and a cysteine in the local microenvironment surrounding the predicted zinc sites (grey spheres). Studies suggest that a typical zinc protease active site contains several histidines coordinating a zinc ion with a catalytic glutamate, and cysteines are also known to coordinate zinc ions.

that the protein shares similar structural features with viral envelope and capsid proteins and forms a decamer with negative surface charge in solution. In our scan, we identified very high scoring sites for the ZINC_PROTEASE.4.GLU.OE1 model at GLU123 in three of the ten subunits (the other subunits did not score above the stringent cutoff). Zinc proteases typically contain three histidines and a glutamate in the active site, with the histidines coordinating a zinc ion and the glutamate acting as the catalytic residue. Our prediction is centered on a glutamate and inspection of the region surrounding it reveals two histidines. We also scanned the structure with a previously published zinc FEATURE model [43], identifying a likely zinc binding site near each predicted zinc protease site.

Because the TOPSAN page suggests a viral origin for 3BJQ and similarity to viral proteins, we reviewed the literature [88, 89] and confirmed that proteases are often involved in the maturation of structural proteins in viruses, cleaving a long polypeptide into functional proteins such as the viral envelope. Furthermore, there

are known examples of auto-catalytic proteases – *i.e.* the protease itself is encoded in the polypeptide which it cleaves to form the mature proteins. It is therefore plausible that 3BJQ could be a zinc protease of viral origin with autoproteolytic capability, a prediction that is compatible with existing analyses of the structure. Even more plausible is the simpler hypothesis that 3BJQ binds zinc in that location, given the presence two histidines and a cysteine, all of which are known to coordinate zinc ions in proteins. The case of 3BJQ, in addition to SeqFEATURE's significant predictions for other TargetDB structures with unknown function, warrants further study.

3.2.5 Protein Data Bank scan results

We additionally scanned every structure in the PDB – about 100 million potential sites – with every SeqFEATURE model. When we consider only those scores that came from models with an AUC of at least 0.95, and were greater than the 99% specificity cutoff defined for that model, 440,460 scores fit these criteria, or about 0.5% of the total number of scores generated. Filtering out redundant scores from proteins with multiple chains results in 298,870 hits in 29,668 structures. The raw data from the scan are available for download [136] on WebFEATURE (see Section 3.2.6); further analysis of these predictions is beyond the scope of this thesis. To access the full library scan for one structure, the user may query by PDB ID; alternatively, one can access all results by querying for a specific SeqFEATURE model.

3.2.6 The WebFEATURE function prediction server

All of the models may be used to scan any protein structure on WebFEATURE, our web-accessible function prediction server (see Figure 3.8) [96, 166]. In particular, users may scan their input structure with the full SeqFEATURE library simultaneously.

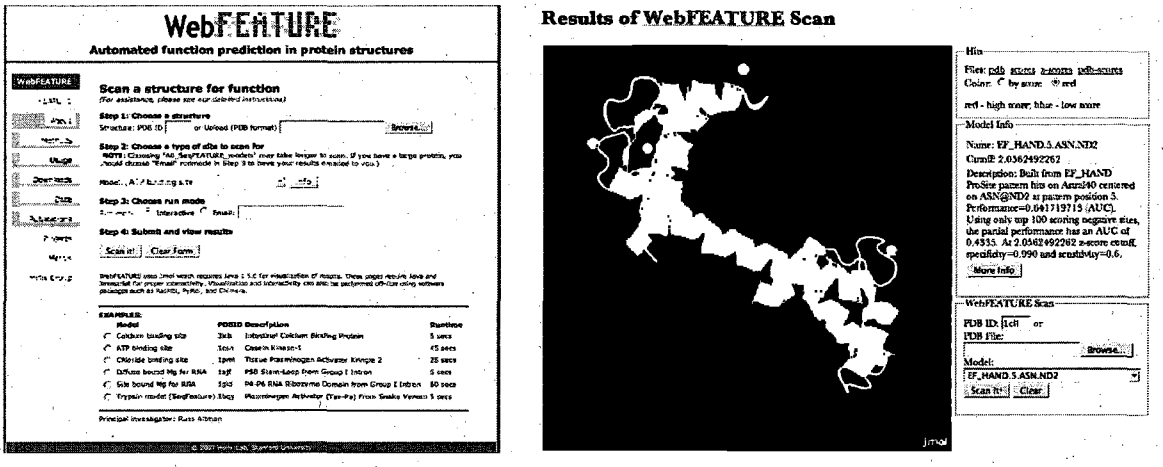


Figure 3.8: Screenshots of WebFEATURE, the web-accessible version of FEATURE. Users may specify PDB IDs or provide their own PDB-formatted files and scan them with any published FEATURE model (a). Scan results may be viewed interactively on the browser (b).

An interactive structure viewer is provided for visualizing scan results. Data from our PDB scan, including predictions on TargetDB structures are also available for download. WebFEATURE is available at `http://feature.stanford.edu/webfeature`.

3.3 Discussion

3.3.1 SeqFEATURE improves over other methods

SeqFEATURE extends earlier work on characterizing functional sites in protein structures by automating training set selection. We have used it to build a library of three-dimensional functional site models, 77% of which have an AUC greater than 0.8. When tested on untrained but known true positives, false positives, and false negatives from their corresponding PROSITE patterns, many models correctly classified all of the true positives and some of the false negatives, and had fewer false positive predictions than the pattern. Even when a model failed to recapitulate every PROSITE true positive, it often correctly classified PROSITE's false predictions.

Furthermore, we show that although SeqFEATURE demonstrates slightly lesser performance than the sequence-based methods overall, it exhibits useful performance trends as sequence identity to proteins of known function decreases. SeqFEATURE, and perhaps structure-based methods in general, should be most valuable in these scenarios, since they sense three-dimensional atomic environments rather than the sequences that fold to create those environments. We observe that this advantage is strongest when the sequence identity is less than 30%, which is well-documented as the 'twilight zone' of sequence analysis [31].

When we further investigate this region of low identity, we see that SSM and 3D templates do not perform as well as SeqFEATURE on the low structural similarity test set. SSM is essentially a fold-matching algorithm, and at low structural similarities the folds of the test structures likely differed significantly from those folds most representative of proteins with the function in question. Theoretically, the 3D template method is more similar to SeqFEATURE, but in reality it performed similarly to SSM. It is possible that the residue triads that 3D templates detect were dependent on exact conservation of sequence features. In contrast, SeqFEATURE was less affected by the reduction in structural similarity because it depends less on specific sequences or arrangements of residues, and instead incorporates abstract physical and chemical properties in a locally defined region.

An important observation is that different methods often complement each other. When we examine the false predictions made by the sequence and structure-based methods, we discover that SeqFEATURE is able to classify a significant fraction of them correctly. This is especially useful at very low sequence identities, where SeqFEATURE corrects over 95% of the positive test examples missed by sequence-based methods. Conversely, cases where SeqFEATURE is incorrect are often recovered by

the other methods. It is not uncommon to see instances where the correct classification is unanimously achieved, but, at the same time, it very rare to have an instance where no method produces the correct classification. These observations underscore the need to have many different types of tools involved in function prediction to build consensus. Because SeqFEATURE uses a microenvironment representation that is neither strictly sequence- nor structure-based, it is uniquely complementary to existing tools.

3.3.2 Challenges in comparing prediction methods

Determining how different methods compare in predicting function is a challenging task, and so neither our procedure for comparing methods nor the interpretation of the results is straightforward. Function itself is broadly defined and does not lend itself easily to simple or computable classification schemes. Many classifications are applicable only to specific types of functions and can differ in the scope of their descriptions, ranging from whole domain labels on sequence (for example, Pfam) to exact locations in structures (for example, SeqFEATURE or 3D templates). Responding to this diversity in description and classification, we made several choices in our comparison of sequence and structure-based methods, each of which carries a certain amount of bias.

In comparing Pfam, HMMPanther, and Gene3D to other methods, for example, we restricted the evaluation to those functions (PROSITE patterns, specifically) whose SeqFEATURE positive training sets mapped unambiguously to the corresponding database assignment. This may have artificially boosted performance of the sequence-based methods, since we, in effect, considered only patterns with very high 'sensitivity'

for each method to begin with based on our training sets. Interestingly, we also investigated HMMTIGR and Superfamily as other methods to include in the comparison, but these tools made very few predictions over the entire set of training and test structures, so we excluded them from the study.

Our choice of gold standard test sites from PROSITE may also be controversial because the test set is limited to those functional patterns that have been manually characterized. In addition, the results may be dominated by a few patterns with many test sites due to the small number of test sites for most patterns. Perhaps most obvious is the high probability that the negative test sites, by virtue of being defined as false positives with respect to the PROSITE pattern, are 'difficult cases'. This means that SeqFEATURE may be predisposed to low specificity, and specificity for all methods overall may suffer because the negative examples are highly similar to the positive examples on at least the local sequence level.

The different types of input used to train each method also have some implications, an important one being that sequence-based methods currently have much more data available to them than structure-based ones. Although this means that the best sequence-based methods currently outperform structure-based methods on our unfiltered PROSITE-based test sets, it does not diminish the need for or value of structure-based methods. Such methods are useful precisely when sequence identity to known proteins is low, as shown in our results on low sequence identity test sets and our analyses on interesting TargetDB predictions.

The two structure-based methods compared here contain an analogous advantage, however, in that they match the query against the entire repository of known protein structures. Thus, if the query has very similar structures (for example, the same protein from different species) in the PDB, SSM and 3D template searches will very

likely result in a high confidence hit to these structures. In cases where the query structure is completely novel, however, SSM and 3D templates are expected to do less well, as suggested by their performance on the low structural similarity test set. SeqFEATURE, on the other hand, because it does not rely on exactly conserved geometries or structural motifs, continues to show robust performance even when the structure does not share significant similarity to known proteins.

Another potential bias may come from limiting the structure-based comparisons to those patterns associated with EC numbers. In order to determine the correctness of predictions from SSM and 3D templates, we required a precise functional classification system. SCOP is a potential alternative evaluation method, but SCOP is a structural classification that does not always map directly to function, so we chose to use EC numbers. This, of course, means that the results of the comparisons may not be representative of how each method performs on non-enzymatic functions. The use of EC numbers is also affected by how accurately and completely the PDB is annotated and by the granularity of function assigned. Several of the test structures on which 3D templates and SSM performed poorly had matches to proteins annotated with only slightly different EC numbers. Thus, 3D templates and SSM should remain valuable tools for gaining insight into the potential function of an uncharacterized protein.

Although the set of patterns and the resulting test sets used here are by no means fully representative or without bias, they enabled us to map our SeqFEATURE models directly to test sets, a non-trivial endeavor given the inconsistency and variety of existing function classifications. It also allowed us to look specifically at where SeqFEATURE improves on or fares worse than the sequence patterns that generated the models. We often chose test sets with biases against our method in order to assess

its operating characteristics accurately; for example, our use of one-dimensional sequence patterns as the gold standard provides a strong advantage to sequence-based methods. Restricting the comparison to patterns that mapped coherently to Pfam, Gene3D, and HMMPanther families may also predispose those methods to good performance. SeqFEATURE exhibited good performance despite these biases.

3.3.3 Advantages of using SeqFEATURE

Because SeqFEATURE focuses on the local microenvironment around functional sites, it can detect function at finer detail than fold-matching algorithms such as SSM. Because it considers both atom-based and physicochemical properties in addition to residue-based ones, it is also capable of generalizing function away from sequence and may be able to detect functional similarities that have converged from different ancestors or that use slightly different residues and a different overall fold to accomplish similar activities. This capability is demonstrated by the fact that SeqFEATURE detects many of the positive examples that the PROSITE pattern misses. The ability to abstract the properties relevant to function independent of sequence or structural homology is one of SeqFEATURE's biggest strengths.

Another one of SeqFEATURE's advantages is that score cutoffs can be adjusted to reflect the user's desired performance criteria – *e.g.* estimated specificity, sensitivity, or positive predictive value. The ratio of true positives to false positives and false negatives is traded off depending on where the score cutoff is set. There are several additional filters one can use to boost the confidence of positive predictions. True hits often manifest themselves as a cluster of high-scoring positive predictions for the same or related functional site models. Single, isolated hits in a protein, although potentially interesting, may not have the exact function represented by the model.

The functional 'fingerprint' of each model (as shown in Figure 3.1) also allows detailed understanding of the physicochemical environment representative of that type of functional site, and detailed inspection of potential positives may boost confidence of positive predictions or help explain the existence of any false positives. Even if the SeqFEATURE prediction is not entirely accurate, the fact that it is based on a representation of the local physical and chemical environment means that we can still make interesting observations about what properties helped the site achieve a high score, and which additional properties may be necessary for the site truly to contain the predicted function.

Most importantly, since SeqFEATURE is not dependent on sequence or overall structural fold, it can be used when either the sequence or the structure is novel. This became evident when we compared the performance of the different methods at low sequence identities and low structural similarities, and found that SeqFEATURE shows a trend to being more sensitive than sequence-based methods at low sequence identities and more sensitive than other structure-based ones at low structural similarities. As shown with the analysis of false predictions and the TargetDB examples, SeqFEATURE is able to predict function where other methods are not. The ability to provide useful predictions on novel structures will become more and more important as structural genomics matures, and SeqFEATURE demonstrates robust performance in this area.

Chapter 4

A review of biological cluster analysis

Although supervised methods allow us to model known phenomena for classification and prediction, it is also important to be able to discover new biological phenomena. In these cases, unsupervised methods that make relatively few assumptions about the underlying structure of the data can help reveal patterns that would be very difficult to detect through manual analysis. Unsupervised methods have been especially prominent in the analysis of high-throughput data sets such as microarrays [139, 154], and more recently applied to databases like the PDB [7, 169]. The data in question typically consist of many-dimensional vectors collected over many biological objects such as genes or proteins. A common goal of these analyses is to identify interesting groups of genes, proteins, or features (*e.g.* motifs, residues, substructures) and then to elucidate the biological relevance of these groups. A general workflow for this type of problem is to define the parameters for comparison (*e.g.* features to compute and distance measures), apply a clustering method, and investigate the resulting clusters for biological significance.

4.1 Clustering algorithms

The most common way to identify interesting subsets of high-throughput data is through clustering [147, 61]. Clustering itself is an unsupervised method that groups objects together based on similarity or distance (see Section 4.2). Most clustering algorithms fall into one of two categories: partitional clustering and hierarchical clustering.

4.1.1 *k*-means clustering and variations

Partitional algorithms determine the entire set of clusters at once, usually through an iterative process. The most widely used partitional method is k-means clustering [100], where k is the number of clusters. The algorithm proceeds as follows:

1. Initialize k cluster centers (often randomly).
2. Assign all data points to the closest cluster center.
3. Compute new cluster centers based on assignments.
4. Iterate until centers or assignments are stable (within some threshold).

k-means clustering is efficient to run on large data sets but the results will differ from run to run due to the random initialization step. Another drawback is the fact that k must be specified beforehand; this can be challenging without prior knowledge of the underlying structure of the data, but heuristics can be used. k-means also provides no measure of how strongly a data point is associated with a cluster.

There are, however, statistical methods that allow points to be associated with clusters with certain probabilities. In mixture modeling, we assume that the data is generated by a number of underlying components, corresponding to the number of clusters we believe is present [51]. Each component is modeled as a distribution

– Gaussian, in the most common case – and similar data points are treated as a random sample from the same component distribution. This means that we can provide estimates of confidence for assignments between data points and clusters.

Much like k-means, finite Gaussian mixture models require specification of the number of clusters. Infinite mixture models avoid this problem by averaging results over all possible numbers of components. We therefore do not need to make assumptions about the number of clusters present in the data set and can produce "soft" clustering assignments [104], which can be much more sensitive to the underlying patterns in the data. But mixture modeling does require estimation of many different sets of parameters, which is typically done using expectation maximization or Gibbs sampling-type algorithms. This can be computationally expensive, especially on high-dimensional data sets [51]. In addition, we are making the assumption that the clusters can be represented as Gaussian random variables, which may be reasonable for gene expression analysis but may not be as suitable in other cases.

4.1.2 Hierarchical clustering

Hierarchical clustering is a process in which the data points are connected based on similarity to form a binary tree. The process can be agglomerative, where individual pairs of cluster objects (data points or sets of data points) are successively merged, or divisive, where the entire set of data points is successively divided; agglomerative methods are more commonly used. The closest pairs of cluster objects are merged into nodes at each step of building the tree. There are many ways to compute the distance between a pair of nodes consisting of multiple data points: the shortest distance (single linkage), the farthest distance (complete linkage), the average distance (average linkage), and distance between cluster centroids (centroid linkage) are the

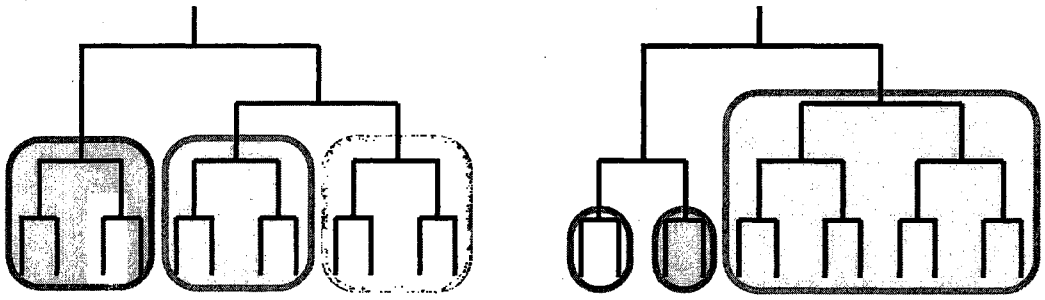


Figure 4.1: Alternative sets of sub-clusters derived from the same hierarchical tree. Any disjoint set of nodes in a hierarchical clustering can be used to define non-overlapping sub-clusters.

most common. Any particular node in the tree defines a cluster as the data points descending from that node. The algorithm is deterministic in that a given data set will always produce the same hierarchical tree using the same distance metric and linkage method; however, it can be computationally expensive to run on large data sets. Another potential problem with hierarchical clustering is that cluster boundaries are ambiguous; subclusters can be defined by selecting any non-overlapping set of nodes, as shown in Figure 4.1.

Other clustering methods, such as self-organizing maps (SOMs), are not discussed here. I refer you to Tamayo *et al* [146] for a brief description of these other methods.

4.2 Distance measures

The most common rationale for grouping objects together is the similarity or distance between their corresponding feature vectors. Clustering algorithms use the computed distance between vectors to decide whether individual objects belong in the same cluster, and also to assess the quality of the clustering. The best distance measure depends on the type of data being used. Distance between binary vectors, for example, is often measured with the Hamming distance, which counts the number of dimensions

in which the vectors differ. A similar metric is the Jaccard coefficient, which is a ratio between the number of features that have a value of 1 and the total number of features that are different or have a value of 1 between the two vectors.

For non-binary vectors, the most common distance metric is Euclidean distance, which represents the Pythagorean distance between the two points represented by the vectors in n-dimensional space. Euclidean distance between two vectors $A = a_1, a_2, ..., a_n$ and $B = b_1, b_2, ..., b_n$ is defined as follows:

$$d = \sqrt{\sum_{i=1}^{n}(a_i - b_i)^2}$$

Large distances mean the vectors are very different from each other. Another popular metric for non-binary vectors is the cosine similarity, which measures the cosine of the angle between two vectors A and B:

$$\text{cosine similarity} = \cos(\theta) = \frac{A \cdot B}{\|A\| \|B\|}$$

Cosine similarity varies from -1 to 1, with -1 indicating that the vectors are exactly opposite, 1 indicating that the vectors are exactly the same, and 0 indicating that the vectors have a random association. Cosine similarity is also known as uncentered correlation, and is equivalent to the Jaccard coefficient when the vectors are binary.

4.3 Cluster evaluation

Evaluating the "goodness" of a clustering algorithm or the quality of a clustering result is often done using external data or through internal measures. The former

involves having a pre-defined set of data where the set of desired clusters or other external knowledge is known and compared to the actual results obtained through the clustering method. Internal measures use the data in the vectors themselves to evaluate how similar the cluster members are and how well separated clusters are from each other. I will refer to these measures as external coherence and internal coherence, respectively. An overall clustering result can also be evaluated by summing or averaging these measures over all of the clusters.

4.3.1 Internal coherence measures

A basic measure of internal coherence is intracluster distance. Generally speaking, intracluster distance calculates how "large" the cluster is based on its constituent data points. There are many variations on intracluster distance, with the most popular incorporating either pairwise distances between all points to all other points in the cluster, or pairwise distances between points and the cluster centroid. The distances may be taken as an absolute value or squared, and the final intracluster distance may be the sum, average, minimum, or maximum of the pairwise distances.

The natural complement to intracluster distance is intercluster distance, or the amount of separation between clusters. Intercluster distance can be calculated using the distances between cluster centers or the distances between closest points. Typically, we may report the smallest intercluster distance for a given cluster (*i.e.* the distance to the closest cluster). We might also be interested in the total intercluster distance for an entire clustering, in which case we would report the sum of the squared intercluster distances between all pairs of clusters. A good clustering will maximize the inter-cluster distance and minimize the intra-cluster distance [18].

One well-known measure that balances both intracluster and intercluster distance

is the silhouette width. The silhouette width for a given point i in a cluster A is defined as follows:

$$S(i) = \frac{b(i) - a(i)}{max(a(i), b(i))}$$

where *a(i)* is the average dissimilarity between i and all other points in cluster A, *b(i)* is the average dissimilarity of i to all other points in the closest other cluster *C*. The quantity *a(i)* is the intracluster distance and *b(i)* is the intercluster distance, so the silhouette width represents how large the cluster and well-separated cluster A is with respect to other clusters. High, positive silhouette values are good and indicate that the given point is close to other points in the cluster but not to other clusters, whereas negative values indicate that the given point is closer to points in other clusters than to points in the same cluster. The silhouette width for an entire cluster can be computed as the average of the silhouette widths for the cluster points.

4.3.2 External coherence measures

Another way to evaluate the quality of a clustering result is to use information known about the objects in the clustering. For example, if the data points are already associated with class labels, we can evaluate how many labels are present in a given cluster and how many members associated with a given label are present in that cluster. *Purity* refers to the frequency of the most common label in each cluster, and is analogous to precision. *Inverse purity*, on the other hand, focuses on the maximum recall for each label. We use the equations for purity and inverse purity [5], where for a set of clusters C and a set of labels L, we define:

$$Purity = \sum_i \left(\frac{|C_i|}{N}\right) max_j \frac{|C_i \bigcap L_j|}{|C_i|}$$

$$Inverse\ purity = \sum_i \left(\frac{|L_i|}{N}\right) max_j \frac{|L_i \bigcap C_j|}{|L_i|}$$

Additional external metrics based on class labels exist, including the Rand statistic and Jaccard coefficient, which consider performance taken over pairs of items (for a review of these and other metrics, the reader is referred to Amigo *et al.* [5]).

4.3.3 Functional coherence using neighbor divergence

Internal coherence measures give some indication of cluster quality but these measures often do not translate into biological relevance. External coherence measures are likely to be more helpful in this regard, but there may not be appropriate pre-defined labels or classifications available. In the case of gene or protein clusters, however, it may be useful to incorporate information from the scientific literature. If there are commonalities in the shared literature associated with members of the cluster, it is likely that there is some degree of *functional coherence.*

Neighbor divergence per gene (NDPG) is one such method for assessing functional coherence of clusters using literature [129, 130]. The algorithm is predicated on the assumption that a group of genes sharing a particular function will have documents in the literature that refer to genes in the group, and documents similar to those will tend to refer to group genes as well (see Figure 4.2). Documents in a gene group are scored based on how relevant their semantic neighbors are to group genes, and the distribution of document scores is compared to a theoretical distribution to produce a functional coherence score. NDPG has been demonstrated to identify functionally important gene groups. It has also been used in concert with hierarchical clustering to determine optimal sub-cluster boundaries [131].

NDPG requires only a corpus of documents and a mapping between documents

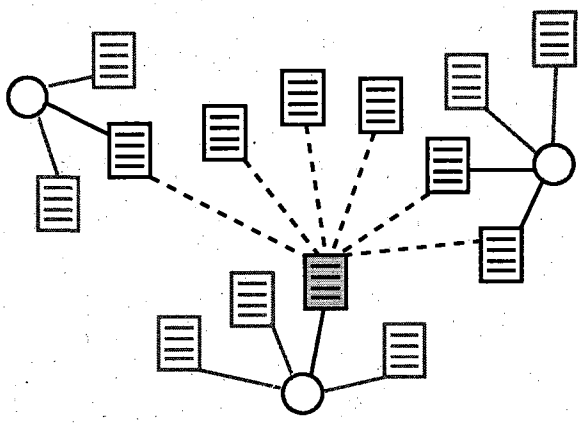


Figure 4.2: Determining functional coherence in biological clusters. Proteins (circles) in a cluster are mapped to documents (solid lines), which, in turn, are associated with a list of similar documents, or semantic neighbors (dashed lines). Each document is scored based on how many proteins mapped to its neighbors are present in the cluster (e.g. green documents, with central document being scored). The distribution of document scores for each protein is compared to a theoretical distribution, and the functional coherence of a cluster is the average divergence between these two distributions across all proteins in the cluster.

and genes (or proteins). Each document is compared to other documents in the corpus to identify similar documents, or semantic neighbors. To do this, we first convert each document into a word vector weighted by inverse document frequency, or *idf*:

$$W_{i,j} = \begin{cases} (1 + \log_2(tf_{i,j}))idf_i & \text{if } tf_{i,j} > 0 \\ 0 & \text{if } tf_{i,j} = 0 \end{cases}$$

$$idf_i = \log\left(\frac{D}{d_i}\right)$$

where $tf_{i,j}$ is the frequency of term i in document j, d_i is the number of documents containing term i, and D is the total number of documents in the corpus. Inverse document frequency is a measure of a word's significance in a background corpus; more common words are less significant than rare words. Similarity between documents is then computed as the cosine of the angle between their two weighted word vectors. The 20 most similar documents are considered the semantic neighbors of the original document.

Note that documents mapping only to the same genes as the original document are excluded from the neighbor list for that document. All documents in the cluster

receive a score based on the fractional references, *fr*, of its 20 semantic neighbors:

$$S_{i,p} = round\left(\sum_{j=1}^{20} fr_{sem_{i,j},p}\right), \text{ where } fr_{k,p} = \frac{n_{k,p}}{n_k}$$

The fractional reference is the proportion of genes the document in question refers to (n_k) that are present in the cluster ($n_{k,p}$). A document's score is thus an integer ranging from 0 to 20, and the scores of all documents for a gene form an empirical distribution. If the genes in the cluster are not related in any way, a Poisson distribution can be used to estimate the theoretical distribution of scores:

$$P(S = n) = \frac{\lambda^n}{n!}e^{-\lambda}$$

where $\lambda = 20 * q$, and q is the fraction of all documents in the corpus that refer to genes in the group.

Kullback-Leibler (KL) divergence is then used to evaluate the difference between the empirical and theoretical distributions of document scores for each gene:

$$D(g||h) = \sum_i g_i \log_2\left(\frac{g_i}{h_i}\right)$$

The average KL divergence across the genes in the cluster represents the functional coherence of the cluster.

4.4 Determining biological relevance

Upon obtaining a clustering result, we typically want to understand the biological basis for those genes or proteins sharing similar behaviors. Traditionally, exploration of

a cluster's potential biological significance has been a manual process. This involves collecting information – usually from the scientific literature – for the individual members of the cluster and then synthesizing all of the disparate pieces into a unifying theme. The process is often convoluted and time-consuming, and so many computational methods for characterizing biological clusters have been developed. These take advantage of the broad array of information now available for many genes and proteins, including annotations in databases and knowledge from the literature.

4.4.1 General methods for cluster annotation

Many cluster analysis tools make use of available annotations such as GO terms, biological pathway assignments, transcription factor binding sites, or functional motifs. Databases like GO, KEGG, and InterPro provide mapping files or query tools to link genes and proteins to annotations. A common approach for identifying biologically relevant information for a cluster, then, is to retrieve the annotations for each gene or protein in the cluster and return those terms or annotations that are significantly enriched. One widely-used scoring method is based on the hypergeometric distribution produced from sampling without replacement from a pool of objects with two possible outcomes. In the case of gene or protein clusters, the hypergeometric function computes the probability of seeing a given annotation or term j times in a cluster of size n given that the term occurs M times over all N possible annotated genes or proteins:

$$p = \sum_{j=x}^{n} \frac{\binom{M}{j}\binom{N-M}{n-j}}{\binom{N}{n}}$$

GO::TermFinder [20], which returns significantly enriched GO terms for a list of genes, is an example of a tool that uses a hypergeometric scoring function.

4.4.2 Literature-based cluster analysis

While database annotations are useful, there are also many methods that use literature directly. Literature as a resource for cluster annotation is attractive for a number of reasons. First, the amount of information present in the literature far surpasses what is currently available in curated databases, and new articles are being indexed in PubMed [126] – the searchable, online database of biomedical literature containing over 18 million abstracts – faster than ever. In addition, the field of natural language processing [102] is mature and yet continually innovating, providing many useful tools for text mining and literature analysis. NDPG is an example of text mining applied to the biomedical domain, and related approaches can help shed light on a cluster's biological details.

Natural language processing (NLP) is becoming more and more popular in biomedicine and has potential uses in information retrieval (identifying documents relevant to a query) and information extraction (identification of assertions or relationships in text) [137, 33, 84]. Text mining tools built on NLP concepts enable researchers to query the literature more effectively, build networks of associations between entities, and even discover hidden relationships [144]. With millions of new scientific articles being published every year, there is a wealth of knowledge waiting to be extracted and utilized. At the same time, it is impossible for manual curation efforts to keep up with the accelerating pace of publication [13]. Biomedical NLP, however, is especially challenging due to the nature of scientific text, with its ambiguous and idiosyncratic terminology, unique uses of punctuation, and often intimidating

sentence structure. Despite these challenges, and the fact that the vast majority of tools fall short of human gold standards on performance for most tasks, it is likely that text mining-based automation will soon be a necessary endeavor.

Text pre-processing

The first step in most text mining tasks is text pre-processing. The text may be filtered for stop words; broken into units, called tokens; normalized into word roots using stemming; and parsed into parts of speech (POS). Stop words are words that appear frequently but contain little information, such as prepositions; these are thus stripped out of free text prior to analysis. Tokenization is a nontrivial task [78] for biomedical text, however, where the non-intuitive use of punctuation and variability in chemical expressions, sequences, and entity names make word and sentence boundaries ambiguous. Tokenizers can be as simple as regular expressions that split up text on whitespace and punctuation, or involve machine learning to train a model for recognizing word boundaries given labeled input.

To avoid redundancy, tokens are often normalized into their word roots, a process called stemming. Suffixes and morphological variants are reduced so that words that are semantically equivalent will be treated as such. For example, 'regulation', 'regulate', 'regulating', and 'regulates' will all become 'regulat' or 'regulate', depending on the stemming algorithm used. Although not specifically optimized for biomedical text, the Porter Stemmer is a widely implemented stemming algorithm [123].

For more complex text mining tasks, it is usually necessary to perform some kind of part-of-speech tagging, where the text is structured and labeled into its syntactic parts. POS taggers tend to be either rule-based models [24] or probabilistic HMM models, trained using known examples or with an iterative bootstrapping approach

[137]. In practice, full parsing is computationally expensive and performance is variable, so shallow parsing – tagging of non-overlapping larger units like noun and verb phrases – is often preferred [137]. POS tagging provides a useful starting point for more complex tasks, such as named entity recognition (NER) [92] or the identification of relationships between entities. NER and relationship identification are extremely helpful for information retrieval and extraction.

Document representation

An important consideration for any text mining task is how the text will be represented. Generally, we maintain documents as units and use the text within them as unique representations of each document. The most common way to represent a document is with a word vector like the one used in the NDPG algorithm (described in section 4.3.3), where the dimensions are the possible words and the values are derived from the frequency of that word in the document [102]. Usually, term frequency is weighted by the *idf* of the term to account for its relative significance in the background corpus. Word vectors are generated after applying pre-processing steps like stop word removal and stemming, and the words themselves depend on the tokenization method. The word vector representation provides a foundation for comparison between documents, ranking of search results, and document classification, although more sophisticated methods may employ more complex representations.

Free text vs. controlled vocabularies

Given the many steps needed to process free text for analysis, much attention has been paid to the use of controlled vocabularies in NLP tasks [143]. Controlled vocabularies consist of manually defined terms representing concepts, as opposed to the

natural language terms one might extract from free text. They are meant to reduce ambiguity between homonyms and synonyms, and can be hierarchically organized to capture 'is-a' and 'part-of' relationships, which can aid in indexing and retrieval. They also obviate the need for the pre-processing steps mentioned above. Examples of controlled vocabularies in biomedicine include Medical Subject Headings (MeSH) [105] and GO. While they are useful for categorizing and annotating scientific literature and biological entities, they must also be maintained with care [32].

Despite the advantages ascribed to controlled vocabularies, there is some debate over their effectiveness. Svenonius [143] reviewed the controversy, noting that some studies showed free text outperforming vocabularies on recall but not precision, and other studies reporting the opposite. It appears, however, that a combined approach using both free text and controlled vocabularies is better than either method alone [63, 135]. For the purposes of cluster analysis, NLP may be employed to identify enriched terms associated with the genes or proteins of interest. In these cases, an additional tradeoff may be interpretability and detail. MeSH might provide terms that are easier to understand, but free text will likely provide more detailed and specific terms which the controlled vocabulary may not be able to capture.

4.4.3 Examples of cluster analysis tools

Cluster analysis is a rich area of study, and many methods are available that address slightly different purposes, both literature-based and annotation-based. But because most tools were developed specifically for data produced by gene or protein high-throughput arrays, they tend to have several limiting characteristics. Most notably, they are predominantly gene-centric, and platform or species-specific. Other cluster analysis tools do not examine the entire cluster at once, but instead enable only

pairwise investigations between members of the cluster. In fact, there are few, if any, tools that can handle arbitrary groups of proteins from multiple species and produce ranked lists of significant, functional terms from diverse knowledge sources, including literature. Such a tool would be extremely useful for our intended purpose of characterizing clusters representing potentially novel functional sites. See Table 4.1 for a survey of tools for biological cluster analysis, with their features and limitations.

4.5 Towards an integrated pipeline for discovering novel functional sites

We now have a number of powerful techniques for exploring many-dimensional data using unsupervised clustering and analyzing the emergent groups using available knowledge. Given the detailed, molecular-level models we can create for protein function using FEATURE, and the increasing number of novel proteins being deposited in sequence and structure databases, the next step is to be able to discover and characterize new biological functions. Yet such a pipeline does not exist.

This may be due in part to the fact that such a pipeline requires the development or application of many disparate methods across many areas of bioinformatics. But as I will describe in the next chapter, we now have a ready source of potentially interesting biological sites – FEATURE microenvironments from the PDB, specifically – which require further analysis. In particular, we need ways to prioritize candidate clusters, break large clusters down into more coherent sub-clusters, and characterize compelling clusters using available knowledge. While some existing methods can be used with

Tool	Description	Proteins?	Species?	Literature?
FatiGO	Identifies enriched GO terms, database keywords, biological pathways, sequence motifs, and transcription factor binding sites in a list of genes.	✗	✗	✗
GO::TermFinder	Identifies significantly enriched GO terms in a list of genes.	✗*	✓	✗
g::Profiler	Detects enriched GO terms, KEGG pathways, and transcription factor binding sites in a list of genes or proteins.	✓	✗	✗
CoPub	Searches the literature for terms enriched in human, mouse, or rat gene lists.	✗	✗	✓
MarmiteScan	Extracts known gene-bioentity co-occurrences from literature for a given list of genes.	✗	✗	✓
GoMiner	Identifies and categorizes GO terms enriched in gene or protein lists.	✓	✓	✗
DAVID	Allows browsing of database annotations and identification of enriched functional terms from arbitrary lists of genes or proteins.	✓	✓	✗

Table 4.1: Comparison of available tools for biological cluster analysis. The criteria on the right refer to whether the tool can handle protein input, input representing multiple species, and uses knowledge from the literature. There are many powerful tools that take advantage of literature, biological databases, and existing annotations to produce lists of enriched terms, pathways, motifs, and other biological features. Since most tools were developed specifically for gene expression data, however, they tend to be gene-centric or stipulate a single organism of origin for the input list. In addition, fewer tools are available that incorporate information from literature. **GO::TermFinder can be modified for protein input by providing a protein-GO association file.*

little or no modification, others are rendered unsuitable based on the nature of the data and our exact needs. As Table 4.1 illustrates, for example, there are few, if any, cluster analysis tools that provide annotation- and literature-based term enrichment for protein clusters consisting of multiple species.

In the next chapter, I describe the methods I have developed to address both the cluster selection and cluster annotation problems, and the functional site discovery pipeline that these methods make possible.

Chapter 5

Discovering novel functional sites

Although many methods exist for recognizing known functions, fewer tools are available for discovering new functions. As we sequence more genomes and solve more novel protein structures, the ability to identify and characterize new functions will become more important. Unsupervised machine learning methods such as clustering can be used to identify interesting biology without prior knowledge. A recent FEATURE study (described briefly in the next section) applied k-means clustering to a representative set of protein microenvironments computed from the PDB with the aim of revealing such interesting groups [169], and active research continues to improve on these methods. Many of the resulting clusters will recapitulate known functions, but others may represent previously uncharacterized functions. Our goal is to characterize microenvironment clusters so that they can be used to train new models of function. To do this, I developed methods for annotating a cluster of proteins with descriptive terms from Swiss-Prot records and PubMed abstracts and adapted and applied techniques to prioritize clusters suitable for annotation.

Much of the work presented in this chapter builds upon the previous work of Yoon *et al.* [169] and Raychaudhuri *et al.* [129, 130, 131]. I am also indebted to Tianyun Liu, who provided the CYS clustering data presented in Section 5.3.4 for the application of the methods presented here.

5.1 Clustering FEATURE microenvironments

As part of a continuing effort to explore protein function through the FEATURE framework, Yoon *et al.* [169] published a preliminary study clustering FEATURE microenvironments. Using a non-redundant subset of the PDB, they computed FEATURE vectors for all amino acids centered on either the beta carbon (hydrophobic residues), or the centroid of polar functional groups (polar residues). Vectors for residues with aromatic rings were calculated at the center of the ring, and a hypothetical beta carbon was constructed for glycines. Prior to clustering, they converted each vector into a binary form to control for the fact that FEATURE vectors normally contain a mixture of data types, which can be difficult to cluster. They also determined that a binary representation resulted in higher quality clusters using training data from 15 SeqFEATURE models.

Using this binary representation, they grouped the approximately 2 million vectors together using k-means clustering. They used a weighted Hamming distance as the distance metric used to evaluate similarity between two vectors. Since k-means clustering requires specification of k beforehand, they experimented with different values of k and found that k=4550 produced clusters that best correlated with functional sites. To analyze the results of this preliminary clustering, they examined all residues that could be annotated with a PROSITE pattern. They found a number of clusters for whom significant representation of one and only one pattern was contained in the cluster. These results suggest that the clustering is able to recapitulate known functions. For more details on the preliminary clustering work, the reader is referred to Yoon *et al* [169].

5.1.1 New developments in clustering microenvironments

The clusters obtained by the published clustering work are by no means definitive. There is much room for improvement in terms of optimal representations for microenvironments, testing of clustering algorithms and implementations, and validation of the approach. For instance, we may be able to improve the clustering results by using microenvironment vectors that have been reduced by principal components analysis, by limiting the clustering to microenvironments that are more likely to be involved in biological function, and by using clustering algorithms that are better suited to the actual space of the data. This is currently an active area of research, and new clusters (as of the time of this writing) are available only for vectors centered on cysteine (CYS) residues.

Unsupervised approaches allow the discovery of potentially novel functions and present unique opportunities for further research and modeling of function. Interpretation and analysis, however, remain challenging. In the second part of this dissertation work, I develop and apply tools for prioritizing, refining, and characterizing clusters so that unsupervised techniques can be better leveraged.

5.2 Methods

5.2.1 Annotating protein clusters

A common problem in biological cluster analysis is interpretation of clusters - that is, determining what biological significance a group of proteins or genes has. A number of methods exist to aid researchers with this problem, but, as discussed in Section 4.4.3, none are well-suited to the characterization of clusters of FEATURE microenvironments. To annotate FEATURE microenvironment clusters, I developed

a system for producing significant terms for an arbitrary list of proteins, incorporating information from PDB files, Swiss-Prot records, and the scientific literature.

Extracting terms from Swiss-Prot, PDB, and PubMed

There are many different types of knowledge available about proteins, and each modality can provide useful clues when investigating a protein cluster. For this reason, we use information from multiple sources, namely PubMed abstracts, Swiss-Prot protein records, and PDB data files. For the purposes of annotating protein clusters, we used version 56.9 of Swiss-Prot, released in March 2009, containing 412,525 protein records. PubMed abstracts were downloaded based on mappings to PubMed identifiers (PMIDs) in the Swiss-Prot records, and PDB data was also downloaded based on the non-redundant set of structures used in the original FEATURE clustering study. We downloaded all data as XML and extracted the desired information for further analysis using the Python `lxml` package [99].

From the PDB files, we extracted labeled ligands (HETATMs) associated with each solved structure. From Swiss-Prot records, we extracted keywords, GO terms, sequence features such as binding sites, subcellular localization information, and protein-protein interactions, in addition to mappings to PMIDs. The annotations were stored only if they were determined experimentally or otherwise verified. PMIDs pertaining to large-scale studies were excluded, as these tend to map to many proteins and contain little functional information. Note that Swiss-Prot is the manually reviewed portion of the larger Uniprot database, and we do not consider records from the unreviewed TrEMBL database. From PubMed abstracts, we extracted the manuscript titles and abstracts as raw text, and the Medical Subject Headings (MeSH

terms) associated with each PMID. The raw text is filtered for stop words and tokenized according to whitespace and punctuation, with hyphenated words treated as a single token. We consider both single tokens (unigrams) and consecutive pairs of tokens (bigrams) as terms. See Table 5.1 for a full list of data considered for annotation purposes.

Table 5.1: Data used in annotating protein clusters.

Database	Data type	Examples
PDB	HETATM	"ZN", "BGC", "SO4"
MEDLINE	Raw text	"bind site", "ptpase activity" , "brca1"
	MeSH term	"Microfilament Proteins", "Tyrosine"
Swiss-Prot	Keyword	"Metal-binding", "Phosphoprotein"
	Subcellular location	"Cytoplasm", "Nucleus"
	Sequence feature	"zinc finger region", "Phosphoserine"
	Interaction	"PIKSR1", "CASP2"
	GO term	"GO:0005515, MF, protein binding", "GO:0006470, BP, protein amino acid dephosphorylation"

Ranking term lists using hypergeometric and entropy-based scoring

To produce ranked lists of the term types mentioned in Table 5.1 (e.g. HETATM, Keyword, GO term, raw text term), we calculate a p-value for each term based on the hypergeometric distribution. This requires first collecting counts for each term in a category over the entire set of Swiss-Prot records. Given a term, we then compute the p-value using the equation presented in Section 4.4.1. We multiply this p-value by the number of terms in that category to correct for multiple hypotheses. We use a corrected p-value cutoff of 0.01 for reporting significant terms in our output.

Since literature-based terms are mapped to PMIDs, they can occur multiple times for each protein, and the distribution of terms across documents and proteins in a cluster may therefore be informative. For example, a term may occur in five documents in a cluster of five proteins, but if all five documents belong to a single protein then the term is likely to be relevant only in describing that one protein. If the five documents belong to different cluster proteins, however, it is more likely that the term is relevant to the entire cluster. In other words, we prefer terms to be evenly or randomly distributed across the proteins in the cluster. To capture this desired quality, we developed an entropy-based scoring function which rewards a random distribution of term-document occurrences across proteins. We adapt the classic entropy formula from information theory into a normalized and weighted score as follows:

$$Score(t) = idf_t \times \frac{S_t}{max(S_)}$$

$$S_t = -\sum_p D_{tp} ln(D_{tp})$$

where D_{tp} is the ratio of the number of documents containing the given term in the given protein p to the number of documents containing the term in the entire cluster. Empirical tests of variations of this scoring function showed that including additional components, such as the term frequency within documents and the fraction of proteins containing the term, did not improve results.

Evaluating scoring function performance on literature-based terms

To test performance of our hypergeometric and entropy-based scoring functions, we devised a set of six test clusters composed of Swiss-Prot proteins associated with specific PROSITE patterns (see Table 5.2). We extracted the PMIDs for each protein and

PROSITE pattern	# of proteins	# of abstracts
COPPER_BLUE	6	18
PROTEIN_KINASE_ST	9	104
ADH_SHORT	10	55
4FE4S_FERREDOXIN	11	45
TRYPSIN_SER	13	128
EF_HAND	19	89

Table 5.2: Test sets used to evaluate scoring functions and functional coherence. We derived six test sets from PROSITE patterns using the data from the SeqFEATURE training sets. These test sets are also used to evaluate the functional coherence method in Section 5.2.2.

the corresponding sets of MeSH and raw text terms for each PMID abstract. We then scored and ranked each set of terms using both hypergeometric and entropy-based scoring. Only literature-based terms were used to evaluate the scoring functions.

Because we suspect that clusters of FEATURE-based protein microenvironments produced by k-means will not be fully coherent, we also tested the scoring functions on clusters containing different proportions of signal to noise. To do this, we created additional clusters that are more dilute than the original test clusters. Since clustering is based on similarity between objects, we used an existing method for finding similar protein microenvironments from the PDB, called S-BLEST [129], to generate lists of candidate proteins from which to dilute our validation clusters.

S-BLEST is a tool that allows retrieval of a ranked list of protein microenvironments similar to a query microenvironment from the rest of the PDB [130]. S-BLEST represents protein microenvironments – centered at specific residue locations in PDB structures – using vectors of physicochemical properties very much like FEATURE. We chose four proteins at random from each validation cluster and used the central functional PROSITE motif residues as input to S-BLEST; each "seed" protein produced a ranked list of similar proteins which we then filtered so that no proteins

contained more than 40% sequence identity to the others or were part of the original cluster.

With an undiluted cluster representing 100% signal, we then successively added proteins from the filtered list to the cluster to make signal percentages of 90, 80, 75, 60, 50, 40, 30, 25, 23, 21, 19, 17, 15, 13, 11, 9, 7, 5, 3, and 1%. For example, to make a cluster with 50% signal from an original cluster containing 10 proteins, we add the 10 most similar S-BLEST proteins from the list. The increased resolution between 25 and 1% is because we expect this range to be a more realistic reflection of the actual clusters we will be attempting to annotate. As a control, we also diluted clusters using proteins drawn randomly from the rest of our background set.

For each validation cluster, we thus created four sets of dilution clusters using S-BLEST results seeded from the four randomly chosen members of the cluster, and four sets diluted using randomly selected proteins. We generated ranked term lists for each diluted cluster in each set using every pairwise combination of data types and scoring algorithms described above as separate experiments. We also input each cluster into a locally installed version of GO::TermFinder to output ranked lists of GO terms. GO::TermFinder is a well-known tool for analyzing gene expression clusters which uses hypergeometric scoring to produce lists of significantly enriched GO terms. It requires a background association file mapping genes to annotations; for our case, we created an association file for proteins using an available Uniprot resource [131].

We computed the F-measure for each experiment on each test cluster using the term lists produced from the original, "100% signal" clusters as a gold standard for truth. F-measure balances precision and recall, and is computed as follows:

$$F - measure = \frac{(1 + \beta^2)(precision \cdot recall)}{\beta^2 \cdot precision + recall}$$

$$precision = \frac{\#\text{ of significant terms that are true}}{\text{total }\#\text{ of terms that are significant}}$$

$$recall = \frac{\#\text{ of significant terms that are true}}{\text{total }\#\text{ of terms that are true}}$$

For the hypergeometric scoring algorithm, we considered a term significant if it had a corrected p-value lower than 0.05. For the entropy-based scoring algorithm, a term is significant if it scores higher than a cutoff. From inspection of the original test clusters, an empirical cutoff of 2.7 produces terms that are clearly functionally related. Calculations were averaged over the four sets of dilutions for each validation cluster and over all validation clusters.

Displaying cluster annotation output

With the multiple types of annotation output the above methods produce, it can be a challenge to make sense of them all. To facilitate exploration of the annotation results, we generate a summary page displaying general information about the cluster along with the top ranked terms in each category. Links from this page lead to external databases (for PDB IDs, HETATMS, and Swiss-Prot accession numbers) or to more detailed pages showing all of the terms scored for each category and the lists of proteins that contributed to each term. The detailed literature annotation pages show the Swiss-Prot proteins and PMIDs associated with each top ranked term, and clicking on a PMID brings up the title and text for that abstract, with an external link to PubMed. See Figures 5.1 and 5.2 for sample screenshots of the output.

5.2.2 Applying functional coherence to protein clusters

When analyzing cluster data, it is desirable to identify clusters which are most likely to produce coherent results. These will be the clusters that contain some signal that

Annotation results for Cluster NODE27X

Cluster information

7 sites in 7 PDB structures, 7 mapped to Swiss-Prot
Functional coherence based on literature: 11.8556 (cutoff for coherence = 3)

Literature annotations based on 68 PubMed abstracts

PDB ID	Site	Swiss-Prot	Protein name
1G4US	CYS481	P74873	Tyrosine-protein phosphatase
1YN9A	CYS119	P24656	Tyrosine-protein phosphatase
1XRIA	CYS150	Q9ZVN4	Probable tyrosine-protein phosphatase At1g05000
1ZC0A	CYS270	P35236	Tyrosine-protein phosphatase non-receptor type 7
1OHCA	CYS314	O60729	Dual specificity protein phosphatase CDC14B
1D5RA	CYS124	P60484	Phosphatidylinositol-3,4,5-trisphosphate 3-phosphatase and dual-specificity protein phosphatase PTEN
2C46A	CYS126	O60942	mRNA guanylyltransferase

Significant annotations (hypergeometric) (p < 0.01)

Sequence features

6.937e-15	Phosphocysteine intermediate
1.121e-10	active site
1.289e-07	Tyrosine-protein phosphatase
0.0006737	Phosphothreonine
0.001403	modified residue
0.008883	Phosphoserine

Keywords

0	3D-structure
0	Hydrolase
2.05e-17	Protein phosphatase
0.0004927	Polymorphism
0.002104	Alternative splicing

Subcellular locations

No significant terms

HETATMS

0.003596	PO4

Interactions

No terms for this category

Significant GO terms (hypergeometric) (p < 0.01)

p (corr)	ID	Type	Description
Full list			
4.495e-14	GO:0004725	MF	protein tyrosine phosphatase activity
1.231e-09	GO:0004722	MF	protein serine/threonine phosphatase activity
1.946e-08	GO:0006470	BP	protein amino acid dephosphorylation
0.0003171	GO:0005634	CC	nucleus

Figure 5.1: **Cluster annotation output – summary page.** We produce HTML output for cluster annotation results with protein identifiers linked to external databases and lists of the top 15 significant terms of each type. In addition to the keywords, subcellular locations, interactions, HETATMS, and sequence features shown here, we also show significant GO terms and literature terms (both MeSH and raw text, ranked both by hypergeometric and entropy-based scoring). Clicking on the type of annotation links to a detailed page showing all terms in that category and the proteins containing that term. For literature annotations, the terms are also linked to the abstracts containing the term.

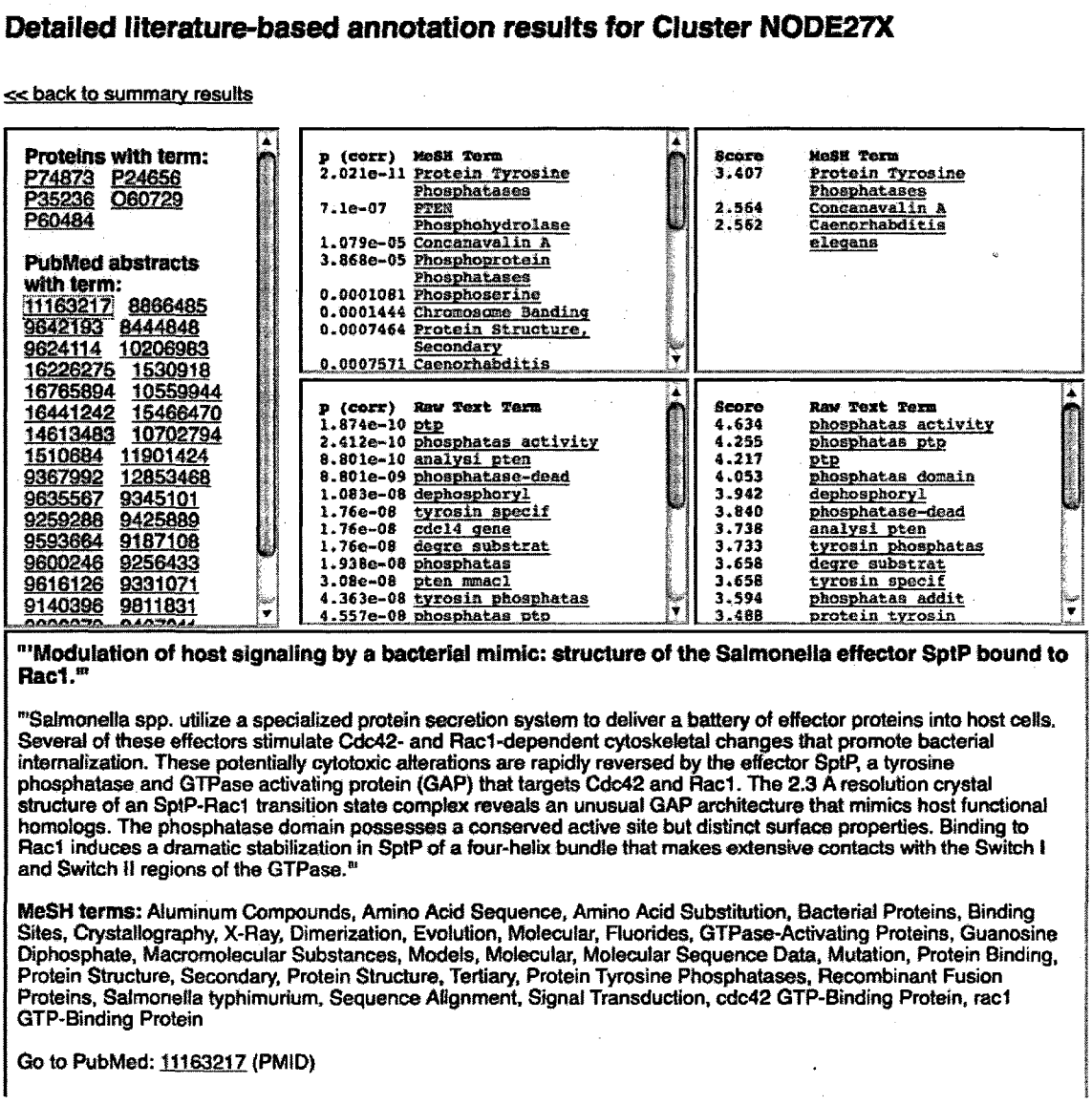


Figure 5.2: Cluster annotation output: detailed literature page. We display the top 15 significant MeSH and raw text terms ranked both by hypergeometric scoring and entropy-based scoring. The terms are linked to information about the proteins and PMIDs containing them, and the PMIDs are linked to the raw text and MeSH terms for that abstract.

the analysis method is designed to detect; clusters in which the external signal is obscure or absent will be much more difficult to validate. To annotate FEATURE clusters, we therefore require methods to determine the coherence of clusters so that we can apply our cluster annotation pipeline effectively. Since FEATURE clusters can be large, having such a method would also allow us to refine FEATURE clusters into more coherent sub-clusters. We adapt the neighbor divergence per gene algorithm for this purpose.

Adapting NDPG to protein clusters

As described in Section 4.3.3, neighbor divergence per gene (NDPG) assesses the functional coherence of gene groups using literature. For our application, we adopt the NDPG algorithm with only a minor change in that proteins are the biological object rather than genes. We determine semantic neighbors for each document as reported by Raychaudhuri *et al.* [129]. We used version 55.4 of Swiss-Prot to map proteins to PMIDs for calculating semantic neighbors and functional coherence.

Although our cluster annotation pipeline uses both MeSH and raw text tokenized into unigrams and bigrams, we used only raw text unigrams to generate word vectors for each document for NDPG. Initial attempts to use MeSH-based word vectors showed that MeSH terms produce word vectors that are not descriptive enough and lead to neighbors that are not truly similar semantically. For example, many abstracts MeSH terms refer to the details of the experimental procedures or materials, meaning that documents with no functional relationship can be computed as being similar. Documents that should be semantic neighbors, on the other hand, can have low similarity due to the fact that the same function or concept can be assigned MeSH terms at different levels in the MeSH hierarchy, such as "Hydrolase" and "Protease".

Given these inconsistencies, we decided to use raw text word vectors to calculate semantic neighbors for documents.

Evaluating the functional coherence metric

For the purposes of exploring the behavior of the functional coherence metric, we used the same test clusters as for evaluating the term scoring functions, and generated additional clusters. To create functional clusters with at least 1000 proteins, we extracted the full list of true positive matches to each PROSITE motif corresponding to our original test clusters. The largest of these was PROTEIN_KINASE_ST with 1303 proteins. We also generated 600 functional clusters of intermediate sizes by randomly sampling 100 times from each of the six full PROSITE clusters, with the restriction that the resulting clusters have at least 10 proteins. To create a corresponding set of completely random clusters, we sampled randomly from the background set of proteins 600 times such that the resulting clusters had between 10 and 1400 proteins.

Because we created our original dilution cluster sets by adding proteins to achieve the desired percent signal, the sizes of each diluted cluster vary greatly, and correspond to different percent signal depending on the size of the original cluster; *e.g.* a diluted cluster of size 50 may represent 20% signal if the original cluster had 10 proteins, or it may represent 60% signal if the original cluster had 30 proteins. Functional coherence is slightly affected by cluster size, so we also created a second set of diluted clusters where the size of the cluster remains constant. To do this, we replaced, rather than added, proteins from the original cluster with either S-BLEST-ranked similar proteins or with random proteins, resulting in diluted clusters of fixed size with varying amounts of signal. We then applied the NDPG algorithm to calculate the functional coherence of all test clusters.

5.2.3 Selecting candidate clusters for analysis

Clusters resulting from k-means can vary widely in size and coherence, and so we need methods first to evaluate the coherence of clusters, and then to break clusters down into more coherent sub-clusters if desired. Since we are primarily interested in clusters that we can characterize using external knowledge, we use functional coherence (Section 5.2.2) as our scoring metric. Based on results from evaluation, we use a cutoff of 3 to designate functionally coherent clusters.

For clusters of reasonably large size (*e.g.* at least 50 vectors), we adapt an approach from Raychaudhuri *et al.* [131] that builds upon the NDPG method. The approach evaluates the functional coherence of nodes in a tree produced by hierarchical clustering. Since cutting a tree at a node produces a sub-cluster from the descendants of that node, we can define an optimal, disjoint set of nodes using a node scoring function, splitting the tree into a set of disjoint sub-clusters. Raychaudhuri *et al.* applied this approach to gene expression data and showed that the resulting sub-clusters represented biologically meaningful groups.

To do the hierarchical clustering step, we used the freely available Cluster 3.0 program [39] which is also part of BioPython, an open source bioinformatics library for Python [28, 38]. I modified the node scoring function to incorporate both internal and external coherence measures to balance physicochemical similarity with available knowledge:

$$S_i = \log_2(n_i) \cdot d_i^2 \cdot f_i$$

where n is the size of the sub-cluster resulting from cutting the tree at node i, d is the node correlation between the two sub-branches merged to produce that node (a measure of internal coherence), and f is the functional coherence of the resulting

sub-cluster. We evaluate each node in the tree, starting at the leaves and ending with the root. We select a node for further consideration if its score is greater than the sum of scores from its selected descendants. Once a node is selected, all of its descendants are deselected. The set of selected nodes at the end of this process represents the set of optimal sub-clusters. For our purposes, we specify a minimum sub-cluster size of 3 during the cluster selection process. Elements not belonging to a sub-cluster with three or more elements are considered singletons and are discarded.

To test whether this cluster selection approach is reasonable, I applied it to a small cluster of 156 microenvironments corresponding to the 15 SeqFEATURE-based training sets (*i.e.* 15 sub-clusters) used to validate the parameters in the published FEATURE clustering study. The microenvironment vectors were normalized by the standard deviation in each feature, and hierarchically clustered using cosine similarity and single linkage.

Exploring parameters for cluster selection

Although the general algorithm for selecting optimal sub-clusters from a large cluster is straightforward, there are a number of parameters to set which depend on the data vectors being clustered and our goals for analysis. These parameters are the normalization used for the vectors, the distance metric used to compare vectors, and the linkage method used for the hierarchical clustering.

The clustering study by Yoon *et al.* [169] used binary feature vectors with a weighted hamming distance, but the change to binary vectors was done primarily to improve computational efficiency. For clustering a smaller number of microenvironments, we decided to use the original microenvironment vectors normalized by the standard deviation in each feature. These can be further compacted using principal

component analysis (PCA) to eliminate redundant features. PCA transforms the original set of features into a set of orthogonal features called principal components. The first principal component explains the most variability in the data, the second explains as much of the remaining variability as possible, and so forth. Since PCA is designed for handling high-dimensional datasets, I investigated whether different numbers of principal components might improve the results of our cluster analysis. For distance metrics, I evaluated cosine similarity and Euclidean distance, and for linkage methods, I tested single, complete, and average linkage.

To assess the suitability of each combination of parameters, I created a larger test set of 1434 microenvironment vectors corresponding to 168 PROSITE patterns from data associated with the published FEATURE clustering study, normalized by the standard deviation in each feature. I generated four additional test sets by selecting the first 80, 40, 20, and 10 principal components. I then applied the cluster selection algorithm described in the previous section to each test set using every combination of [cosine similarity, Euclidean distance] $\times$ [single linkage, complete linkage, average linkage].

There are several ways to evaluate the results of the cluster selection algorithm, using both internal and external quality measures as described in Section 4.3. I computed % coverage, *i.e.* the fraction of input vectors contained in the resulting set of optimal sub-clusters, and the average silhouette width for each sub-cluster as internal measures. For external quality, I calculated the precision and recall as described in Section 4.3.2.

5.2.4 Application to FEATURE clustering data

I applied the cluster selection and annotation methods to a new, unpublished set of clusters produced from a k-means clustering of microenvironment vectors centered on the beta carbon of cysteine (CYS) residues and reduced to 80 principal components. Based on results from the parameter evaluation, I used cosine similarity with single linkage to perform the cluster selection step on the CYS clusters. For the purposes of analysis, I generated significant annotations using the methods described in Section 5.2.1 only for clusters or sub-clusters with functional coherence >3.

5.3 Results

5.3.1 Evaluation of literature-based scoring functions

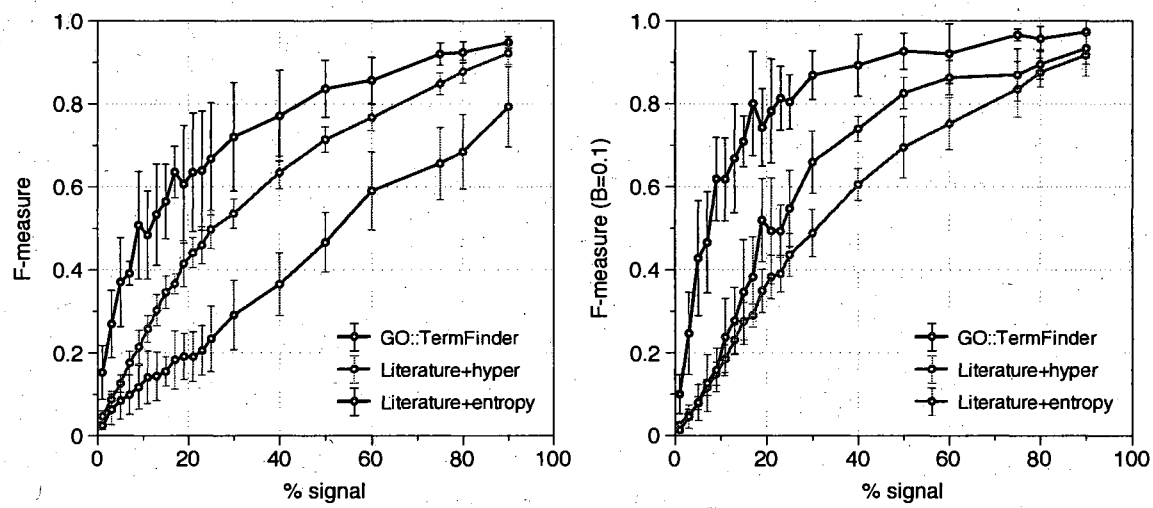
We applied the hypergeometric and entropy-based scoring functions to raw text and MeSH terms associated with six test clusters derived from PROSITE patterns. Sample term lists are shown in Table 5.3. Both scoring functions clearly are effective at ranking relevant terms for both raw text and MeSH given a high-signal protein cluster. There are some interesting things to take into account about the different term types and scoring methods. First, many more raw text terms are produced for each cluster than MeSH. Second, MeSH terms tend to be more coarse-grained, whereas raw text is able to identify interesting terms such as catalytic residues. Also, because entropy-based scoring takes into account the distribution of terms across documents as well as across proteins, it tends to be more discriminatory and produces fewer significant terms given a reasonable score cutoff. For the purposes of evaluation, we used a score cutoff of 2.7, which was empirically determined using the six original test clusters.

	Hypergeometric	Entropy-based
MeSH	Cyanogen Bromide	Cyanogen Bromide
	Factor IX	Factor IX
	Serine Endopeptidases	Serine Endopeptidases
	Factor X	Aspartic Acid
	Complement Activating Enzymes	Thrombin
	Complement Pathway, Alternative	Epidermal Growth Factor
	Blood Coagulation Disorders	Trypsin
	Pancreatic Elastase	Serine
	Factor VII	Structure-Activity Relationship
	Epidermal Growth Factor	Peptide Hydrolases
Raw text	serine proteas	proteas zymogen
	serin proteinas	ser-195
	resolut	asp-102
	chymotrypsin	his-57
	serin	serin proteinas
	proteas zymogen	chymotrypsin
	asp-102	residu factor
	ser-195	c factor
	crystal	asp-102 ser-195
	his-57	zymogen

Table 5.3: Sample term lists for the TRYPSIN_SER cluster. The top 10 MeSH and raw text terms ranked by hypergeometric and entropy-based scoring are shown. Note that MeSH terms tend to be more coarse-grained than raw text, which is able to identify catalytic residues such as "ser-195".

To test the behavior of the scoring functions on more realistic clusters, we created a series of noisy clusters by adding structurally similar and random proteins to the original test clusters, and then generated list of significant literature terms using both scoring functions. We also generated term lists using GO::TermFinder, with a protein-GO annotation file instead of the default gene-GO annotation file. The gold standard in each case was the list of significant terms corresponding to the original test clusters. Figure 5.3 shows the F-measure as a function of % signal for GO::TermFinder, hypergeometric scoring with literature terms, and entropy-based

scoring with literature terms, using two different values for β. Since we are concerned mostly with how the scoring functions behave relative to the amount of functional signal in a cluster, we do not draw any conclusions about the relative performance of the different scoring methods to each other. Instead, we observe that all methods perform very well when the % signal is high, and all methods exhibit a steep drop-off in performance below about 40% signal, especially when we consider precision as more important than recall ($\beta = 0.01$). This drop-off is mirrored in the actual term lists, as shown in Table 5.4.



(a) $\beta = 1.0$ **(b)** $\beta = 0.01$

Figure 5.3: F-measure of all scoring methods drops off below 40% signal. All calculations are based on the original (100% signal) clusters as the gold standard, and calculations were averaged over the four dilution sets generated using S-BLEST matches. When we weight precision and recall equally (a), hypergeometric methods show better performance. When we weight precision more than recall, entropy-based scoring outperforms hypergeometric scoring on literature terms (b). All methods show poor performance below 40% signal, suggesting that the amount of functional signal in a cluster is the most important factor for effective cluster annotation.

100% signal	75% signal	25% signal
proteas zymogen	proteas zymogen	proteas zymogen
ser-195	*replac method*	*s1 specificity*
asp-102	ser-195	*1 resolut*
his-57	11	asp-102
serin proteinas	his-57	asp-102
chymotrypsin	*insert loop*	*pocket*
residu factor	serin proteinas	*solv*
c factor	zymogen	*cleft*
asp-102 ser-195	chymotrypsin	his-57
zymogen	residu factor	*structur complex*

Table 5.4: Degradation of term list coherence with decreasing functional signal. Shown here are raw text term lists for the TRYPSIN_SER test cluster at 100%, 75%, and 25% signal. We see that a conceptually vague terms (in italics) appear at 75% signal, and many more appear at 25% signal. Given that this cluster is somewhat idealized, we can imagine that the term lists derived from actual data would be even noisier. It is therefore important to ensure as much functional coherence in a cluster as possible before attempting to annotate it.

5.3.2 Evaluation of the functional coherence metric

We compared the functional coherence of random protein clusters and clusters associated with the six PROSITE patterns used for testing the literature scoring functions. The functional clusters ranged in size from six proteins to over 1300 proteins. As shown in Figure 5.4, functional clusters attain much higher functional coherence scores than random clusters. We also calculated the functional coherence of the dilution clusters to see how the amount of signal in a cluster affects its functional coherence (see Figure 5.5). Functional coherence clearly decreases as % signal decreases; when the size of the cluster is fixed, the relationship is approximately exponential.

When cluster size is not fixed, however, we can see a slight increase in functional coherence at very low % signal. This likely results from the sharp increase in cluster size at very small percentages when the dilutions are additive; for random clusters, we observed that functional coherence increases very slightly with increasing size. Our

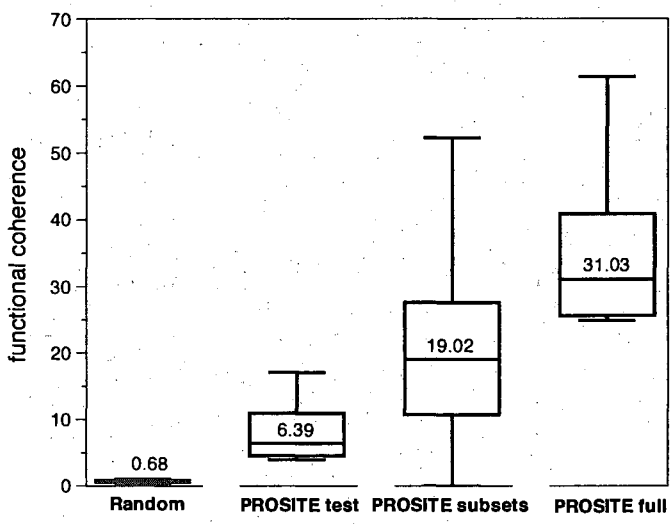


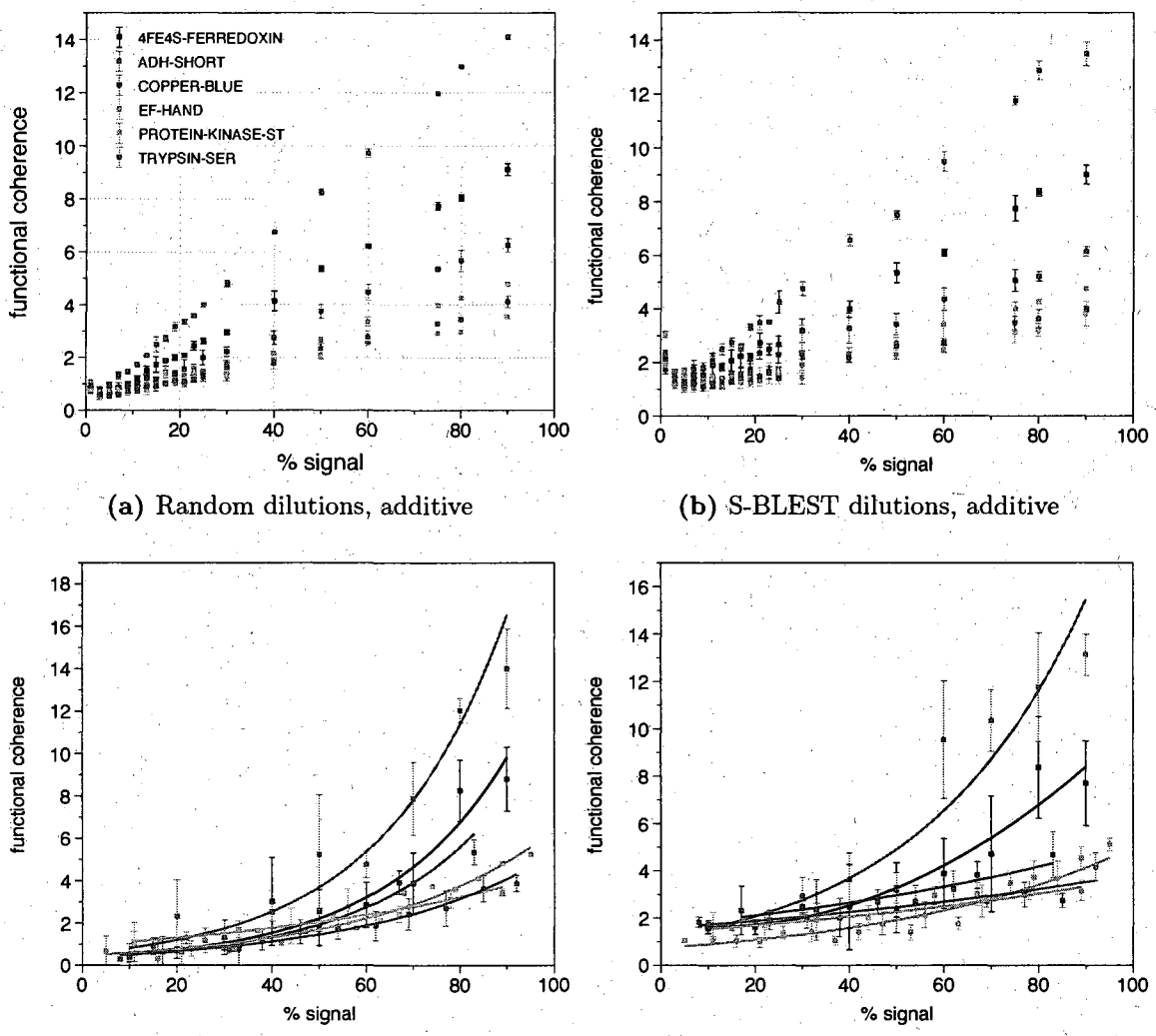
Figure 5.4: Functional coherence of random and functional clusters. We calculated functional coherence for the original PROSITE test clusters, the full versions of the test clusters (*i.e.* all proteins matching the PROSITE pattern), and random subsets of the full clusters, as well as completely random clusters varying in size from 6 to 1400 proteins. Random clusters have a median functional coherence of 0.68, which is much lower than the median functional coherence of the PROSITE test clusters.

applications, however, will be limited to clusters small enough that we can ignore this effect. Based on these observations, we can set an empirical cutoff to distinguish functional clusters from non-functional clusters.

5.3.3 Evaluation of the cluster selection approach

To see whether the cluster selection approach is reasonable, we applied it to a small test set consisting of 156 microenvironments corresponding to 15 SeqFEATURE models. Using cosine similarity and single linkage for the hierarchical clustering step, and a minimum sub-cluster size of 3, we recover all 15 of the original clusters as distinct sub-clusters (Figure 5.6), although two – ZINC_PROTEASE and ADH_SHORT – were each split into two sub-clusters. The original cluster for ADH_SHORT actually consists of two types of microenvironments – one centered on the hydroxyl oxygen in the sidechain of the active site tyrosine, and the other on the aromatic ring of the tyrosine. The two sub-clusters for ADH_SHORT consist exactly of these two types and they are immediate neighbors in the hierarchical tree.

Since our goal is to produce clusters with better signal-to-noise, well-separated



(a) Random dilutions, additive

(b) S-BLEST dilutions, additive

(c) Random dilutions, fixed

(d) S-BLEST dilutions, fixed

Figure 5.5: Functional coherence of diluted clusters. We calculated functional coherence for clusters diluted by adding random proteins (a) and structurally similar proteins (b), and for clusters diluted by replacing cluster proteins with random (c) or structurally similar proteins (d). In each case, functional coherence clearly decreases as % signal decreases.

clusters with high purity are desirable. There are a number of parameters we can modify, including the distance metric and linkage method used for hierarchical clustering, and the degree of normalization and principal components of the microenvironment vectors. We investigated all combinations of [cosine similarity, Euclidean

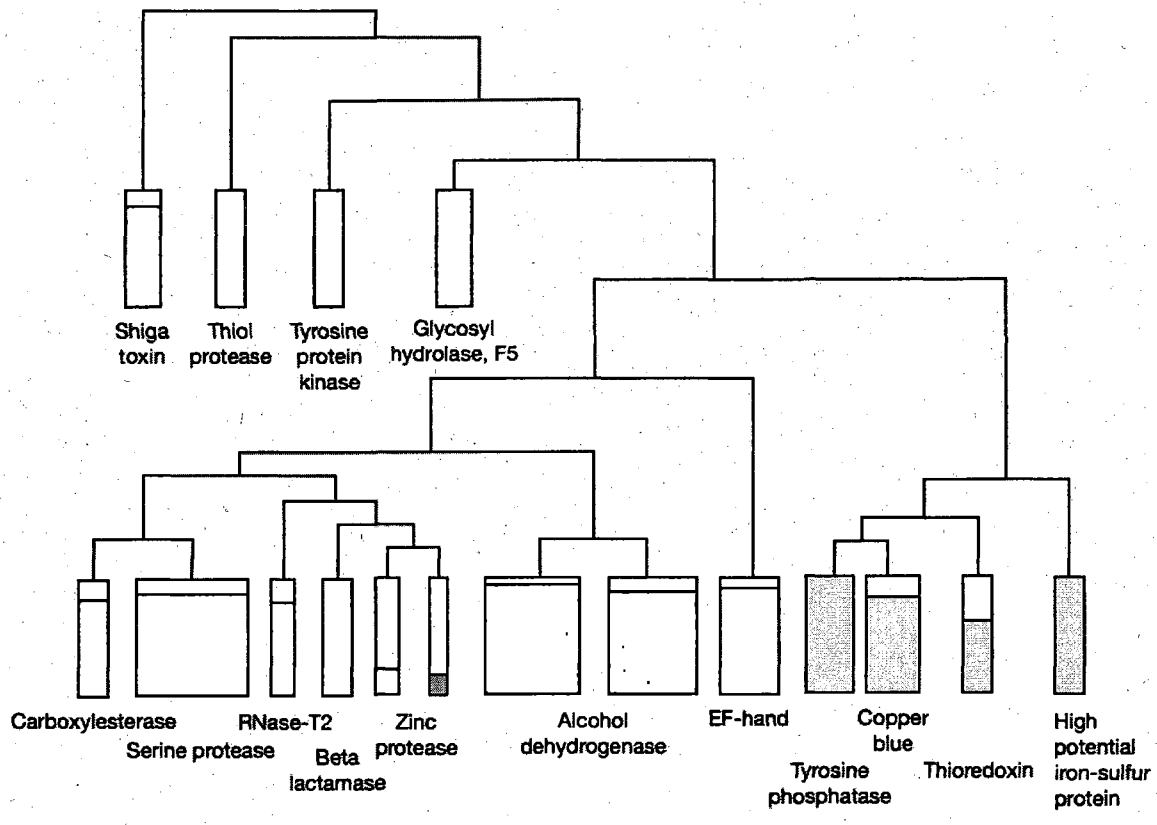


Figure 5.6: Approximate tree of sub-clusters selected from the 15-model test set using cosine similarity and single linkage. The tree shows relative placements of sub-clusters in the hierarchy (branch lengths not to scale). The width of the boxes represents the size of the sub-cluster, and the proportion of the box that is colored represents the proportion of the labeled model captured by that sub-cluster. For example, the carboxylesterase sub-cluster has 5 microenvironments, which represent 85% of the total microenvironments for carboxylesterase present in the test set. This figure shows that our cluster selection approach is able to redefine the basic separations present in the test set.

distance] + [average linkage, complete linkage, single linkage] + [no PCA, 80, 40, 20, and 10 principal components] on a larger test set of 1434 microenvironments mapped to 168 PROSITE patterns. For each combination, we determined the % of the test set captured, plotted the distribution of silhouette widths (Figure 5.7), and calculated the purity and inverse purity of the resulting sub-clusters (Figure 5.8).

In general, cosine similarity produced better silhouette widths and higher values

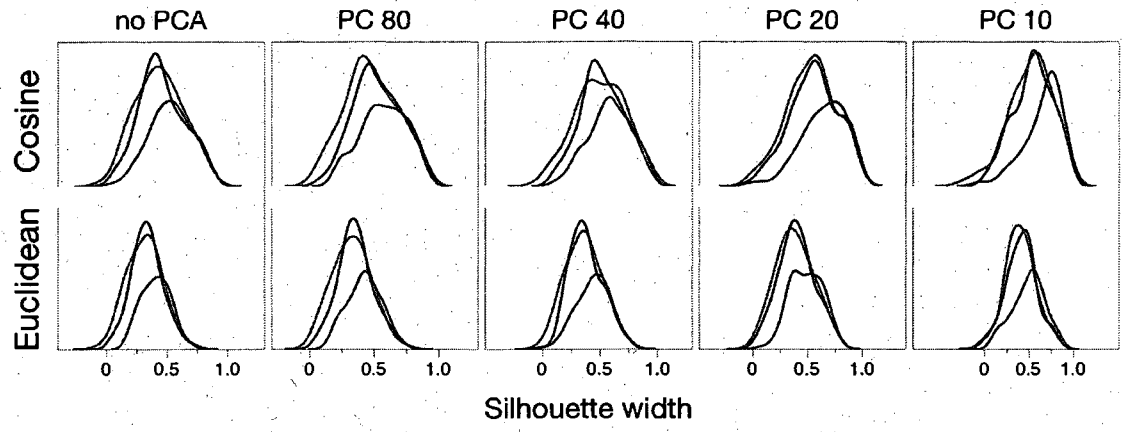


Figure 5.7: Distributions of silhouette widths for combinations of parameters. We plot the distribution of silhouette widths for sub-clusters derived from each principal component data set hierarchically clustered using both cosine similarity and Euclidean distance with average (black), complete (red), and single (blue) linkage. Cosine similarity produces better silhouette widths than Euclidean distance. Single linkage outperforms average and complete linkage with regard to purity and silhouette widths. Silhouette widths also improve with the use of fewer principal components.

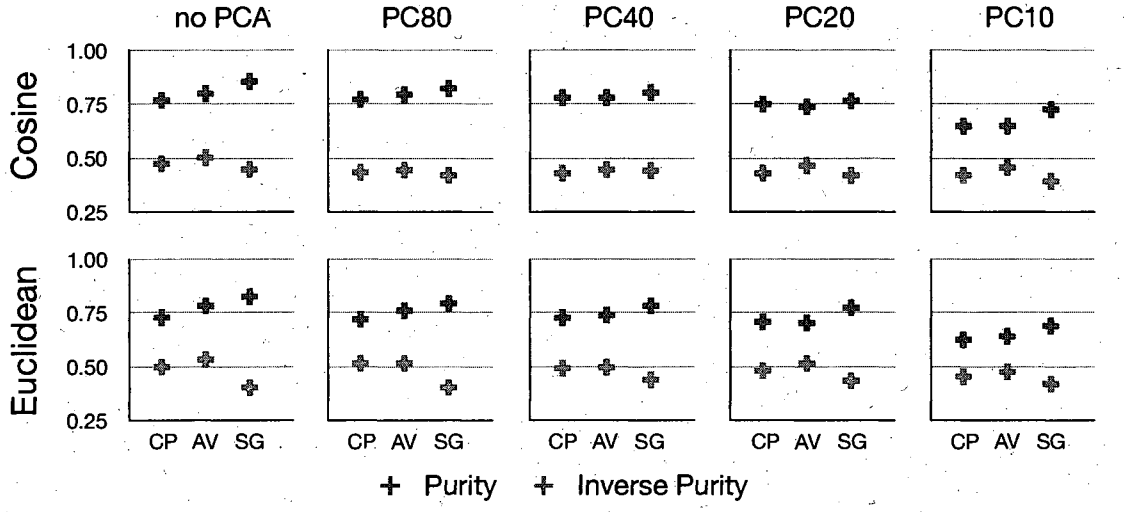


Figure 5.8: Purity and inverse purity for combinations of parameters. We calculated purity and inverse purity for sub-clusters derived from each principal component data set hierarchically clustered using both cosine similarity and euclidean distance with complete (CP), average (AV), and single (SG) linkage. Cosine similarity produces sub-clusters with higher purity but Euclidean distance produces sub-clusters with better inverse purity. Performance decreases as fewer principal components are used. Single linkage results in the lowest inverse purity for each set but provides the highest purity.

for purity and inverse purity. Decreasing the number of principal components also produces better silhouette widths, but using no PCA produces better purity and inverse purity. Purity and inverse purity decrease steadily as fewer principal components are used. In addition, although single linkage captures much less of the test set and suffers from poor inverse purity, it produces sub-clusters with the highest purity. Given that our goal is to recapture or discover functional sites with good reliability, we value purity above all other considerations. Cosine similarity, single linkage, and no PCA are the parameters that perform best with regards to that goal. Note that the choice of distance metric and normalization method also affects the initial k-means clustering itself, and so the choices of parameters for clustering and cluster selection, although ideally identical, might necessarily be different to achieve the best results in each process.

5.3.4 Application to FEATURE clustering data

We applied cluster selection as described in Section 5.2.3 using cosine similarity and single linkage to 40 unpublished, cysteine-based (CYS) clusters; these microenvironment vectors were normalized to 80 principal components for the purposes of k-means clustering. All sub-clusters with at least five microenvironments were considered for further analysis, since five is the minimum number of sites we have used for training FEATURE models in the past. We chose not to analyze the clusters produced from the work done by Yoon *et al.* since there was evidence that the newer approaches produce clusters of higher quality and more reasonable size.

From the 40 CYS clusters, the cluster selection method produced 218 optimal sub-clusters with more than 5 microenvironments. To prioritize analysis, we focused on sub-clusters with functional coherence scores >3 (70 sub-clusters, see Appendix

C.2 for a full list), and sub-clusters with high internal node correlation from the hierarchical clustering results. We applied the annotation methods described in Section 5.2.1 to these sub-clusters.

When we examined sub-clusters with low functional coherence but the highest internal correlation, we found that many were associated with structural artifacts such as disulfide bonds, very exposed regions, or the presence of alternate coordinates. Clusters 9 and 26 seem to consist predominantly of these types of sub-clusters. Upon examining clusters with higher functional coherence, however, we see that they also have emergent themes, but this time of a functional sort. Clusters 32 and 33 pertain to zinc-binding, clusters 22 and 23 are heavily annotated with cytochromes, and cluster 30 contains iron-binding sub-clusters. Although the cytochrome-associated sub-clusters are only found in clusters 22 and 23, sub-clusters related to metal ion-binding, phoshatase, and kinase activity are found in multiple clusters.

Since cysteine residues are often involved in binding metal ions, it is unsurprising to see many sub-clusters with metal-binding as the dominant functional annotation. We were, however, intrigued by the fact that they did not group into the same k-means cluster. To investigate whether k-means was partitioning the clusters accurately, we combined 15 zinc-binding-associated sub-clusters belonging to four clusters into one large cluster and ran it through the cluster selection process again. The exact same sub-clusters were produced (excluding two microenvironments from one sub-cluster that were deemed singletons in the new result), indicating that the cluster boundaries from k-means are reasonable within the parameters given.

Although many of the zinc-binding sub-clusters differ according to their coordination types – 2 CYS and 2 HIS or 4 CYS, for example – many seem to bind zinc in the same manner. When we examined the sets of principal component vectors for

sub-clusters with identical coordination types, we confirmed that there are indeed significant differences between them. Therefore, while the coordinating residues are identical, there are more visually subtle ways – *i.e.* specific principal components – in which they differ. See Appendix C.1 for more details on our zinc sub-cluster analyses.

Further analysis of some of these functionally coherent sub-clusters yielded encouraging results. Sub-cluster 27 in cluster 21 (Clust21-Sub27; sub-clusters are numbered by the node to which they correspond in that cluster's hierarchical tree) represents the active site of tyrosine protein phosphatases. Each central CYS is also annotated as the active site residue in that protein's Swiss-Prot record. Clust33-Sub49 represents a copper-binding site, with the majority of its member proteins belonging to the blue copper family (see Figure 5.9) of cyanins. One of the structures is bound to zinc rather than copper, but is known to bind copper in that location. All other structures in Clust33-Sub49 are bound to copper. The microenvironment contains two HIS residues helping to coordinate the ion, and a MET residue, which is not always bound but is always nearby. Terms associated with copper-binding and electron transport dominate annotations for this sub-cluster.

Another copper-binding sub-cluster (Clust1-Sub13, see Figure 5.10) is in an entirely different cluster, and this environment seems to be associated with the family of multicopper oxidases. Again, all structures are bound to copper through the central CYS residue, in addition to two HIS residues. In three out of the five microenvironments, a MET residue is present but not bound. Like above, the annotations center around copper-binding, but with keywords for "oxidoreductase" rather than "electron transport", distinguishing the function of this sub-cluster from that of Clust33-Sub49.

Interestingly, both of these copper-binding sub-clusters correspond to the same type of copper center – type 1, which is coordinated by CYS, two HIS residues and

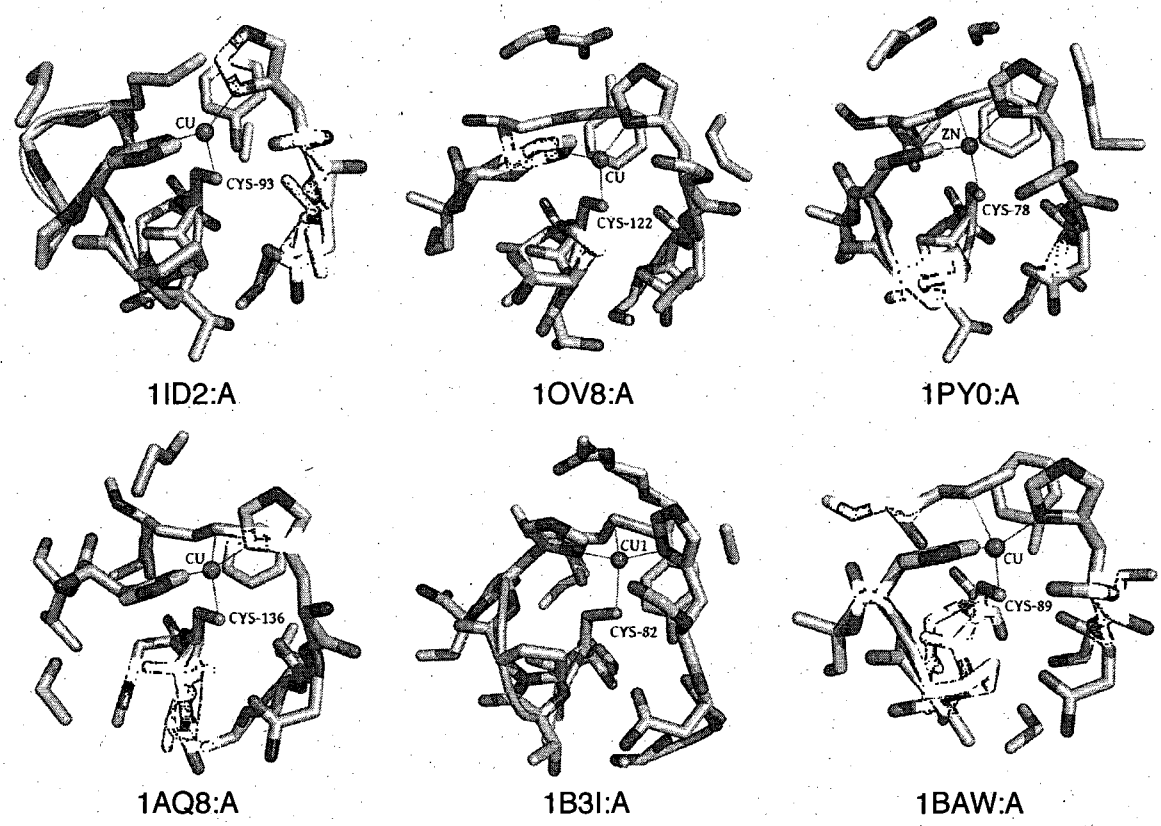


Figure 5.9: Clust33-Sub49 – Blue copper protein-associated copper-binding sites. Clust33-Sub49 is a copper-binding sub-cluster associated with blue copper type proteins. Copper ions present in the structure are shown; 1PY0 contains zinc instead of copper, though it is known to bind copper in that location. Coordinating residues are indicated with connecting lines. The central CYS is colored green and labeled.

a fourth residue [67]. In plastocyanins, the fourth residue is a MET, while in multi-copper oxidases it is often substituted by a non-coordinating residue [106]. This is consistent with our observations in these two sub-clusters. Another interesting observation is that structure 1V10:A in Clust1-Sub13 is thought to have copper oxidase function based on other computational predictions; our grouping of it together with other copper oxidases supports this prediction.

In addition to the previous examples, we also identified sub-clusters representing conserved environments in protein kinases and cytochrome C proteins, as well as

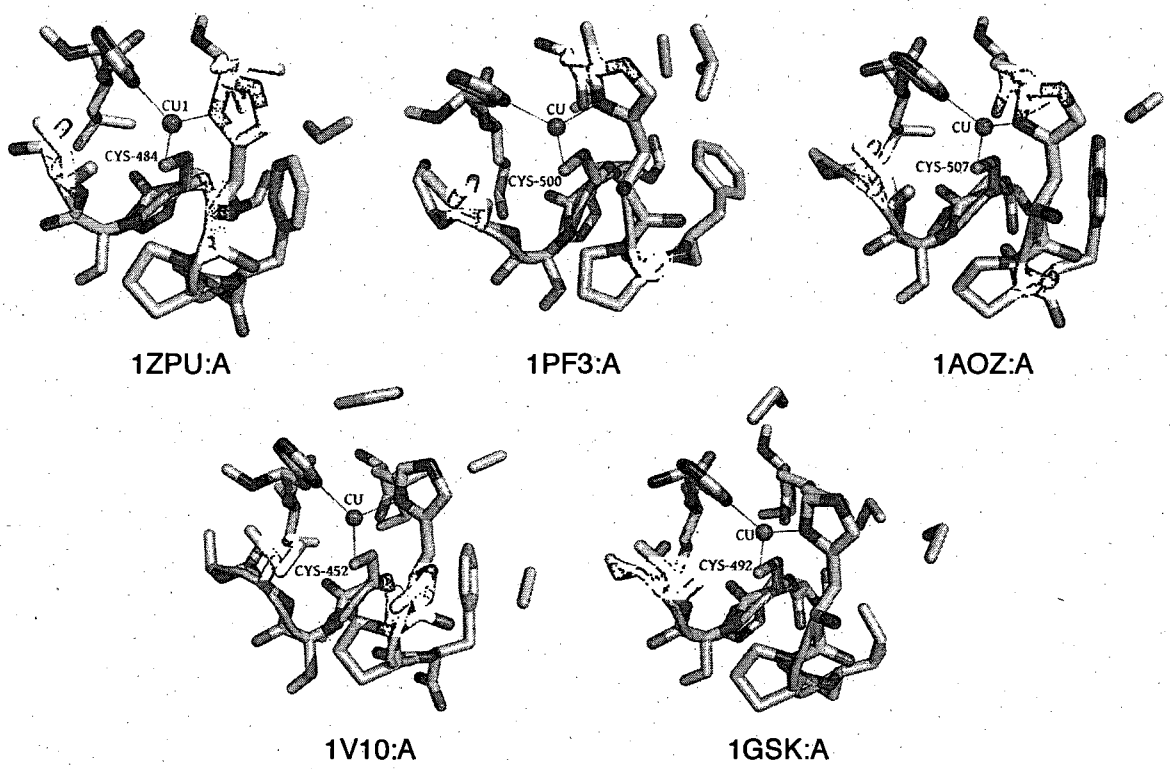


Figure 5.10: Clust1-Sub13 – Multicopper oxidase-associated copper-binding sites. Clust1-Sub13 is a copper-binding sub-cluster associated with multicopper oxidases. Copper ions present in the structure are shown with coordinating residues indicated with connecting lines. The central CYS is colored green and labeled.

iron, iron-sulfur, and zinc binding sites. Zinc binding is particularly interesting, as there are many motifs and catalytic sites known to bind zinc [6]. Figure 5.11 shows four types of zinc binding sites present in distinct sub-clusters in our data set. The first three types are mononuclear, where a single zinc ion is coordinated by different numbers of CYS and HIS residues – 4 CYS, 3 CYS and 1 HIS, or 2 CYS and 2 HIS. Zinc-binding of this type is typically for protein structural stability. The fourth type shown is a cocatalytic dinuclear zinc site coordinated predominantly by HIS residues and a water molecule. These types of sites are found in metalloenzyme active sites, where the zinc ion is required for catalytic activity.

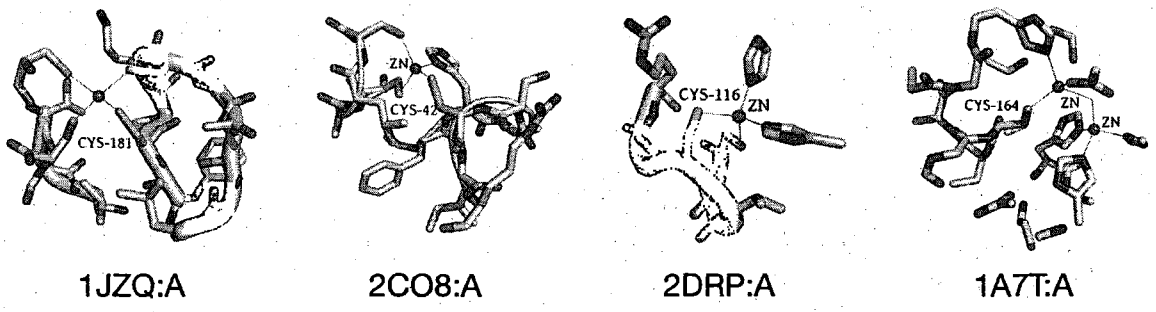


Figure 5.11: Representative microenvironments from four distinct zinc binding sub-clusters. From left to right, the representative sites are from Clust32-Sub222, which binds zinc with four CYS residues; Clust1-Sub118, which binds with 3 CYS and 1 HIS residue; Clust33-Sub156, which binds with 2 CYS and 2 HIS residues; and Clust1-Sub53, where one or two zinc ions are coordinated by the central CYS, a number of HIS residues and a water molecule.

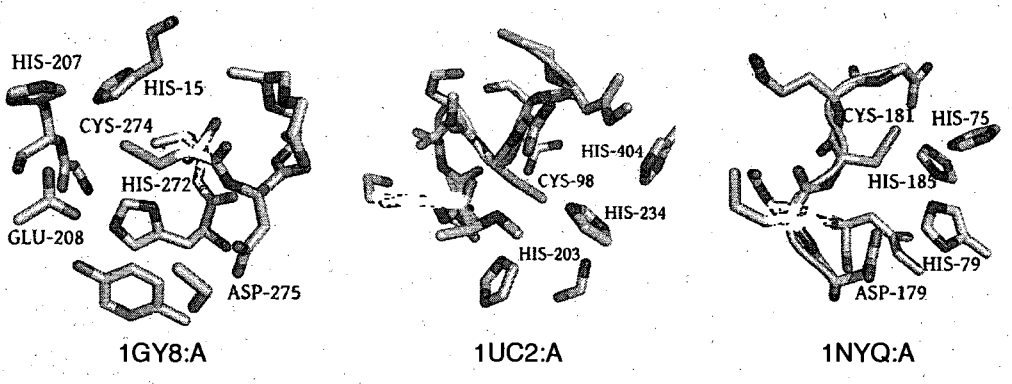


Figure 5.12: Predicted zinc binding sites in Clust1-Sub53. All of the sites contain three nearby HIS residues which could potentially coordinate a zinc ion along with the central CYS residue (green). 1GY8 contains nearby ASP and GLU residues, which are also known to coordinate zinc.

Further inspection of zinc-binding sub-clusters yields some interesting findings. Structure 1V7O:A in Clust1-Sub52, for example, is the apo form of a zinc-binding protein, with our predicted site at CYS116 corresponding to the known zinc binding site. In addition, several proteins in Clust1-Sub53 have not been proven to bind zinc at the sites specified (CYS181 in 1NYQ:A, CYS98 in 1UC2:A, and CYS274 in 1GY8:A), but have microenvironments highly suggestive of zinc binding. The salient features include the presence of several HIS residues and occasionally an ASP or GLU residue around the central CYS (see Figure 5.12). There is evidence, however, that 1NYQ and

1UC2 may bind zinc at those locations. Others have noted the presence of conserved HIS and CYS residues in 1UC2 corresponding to those in our site, similar to zinc metalloenzymes and tRNA synthetases [114]. 1NYQ, a threonyl-tRNA synthetase, is already known to bind zinc [152], but in the crystal structure zinc is bound at a location far from our site. CYS181 may thus be a novel zinc binding site for 1NYQ. The third protein, 1GY8, is a UDP-galactose 4'-epimerase from *T. brucei* [138] that is not known or suspected to bind zinc.

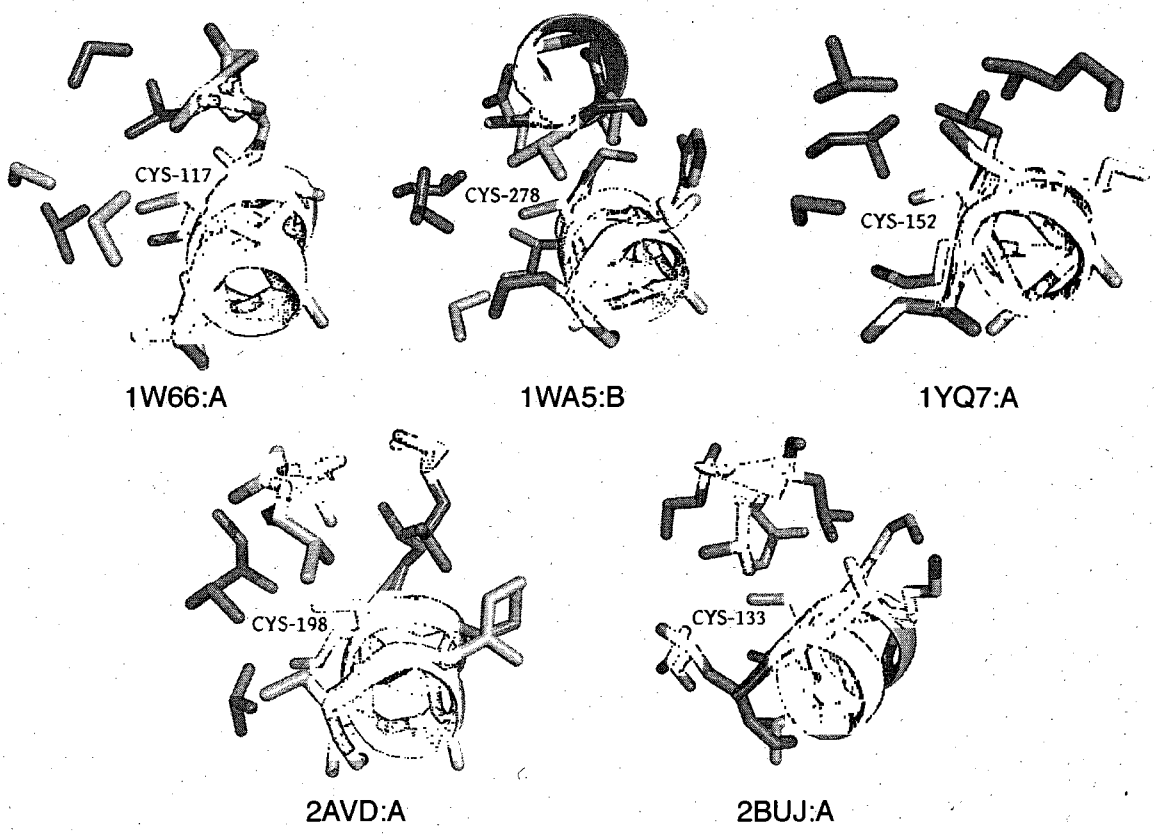


Figure 5.13: Clust8-Sub25 – A potential structural motif. In Clust8-Sub25, the central CYS (labeled and shown in green) is part of an alpha helix, and its sidechain is surrounded by numerous aliphatic, hydrophobic residues such as ILE, LEU, and VAL (shown in purple). This microenvironment may have a structural role due to its lack of reactive chemical groups and recurrence across diverse proteins.

In addition to cases where the prediction is clear based on other members of the sub-cluster, we also have cases where the theme of the sub-cluster is more obscure. Consider, for example, Clust8-Sub25 (see Figure 5.13). This sub-cluster has eleven microenvironments, all of which are characterized by an alpha helix containing the central CYS residue, whose sidechain is surrounded by an abundance of hydrophobic, aliphatic residues such as ILE, LEU, and VAL. Since the microenvironment is not usually surface-exposed, it likely not associated with an explicit function. The chemically neutral makeup of the microenvironment, however, as well as its recurrence across diverse proteins, indicates that it may have an important structural role.

Another intriguing example is Clust5-Sub70 (see Figure 5.14). This sub-cluster contains 12 microenvironments, eight of which are from protein tyrosine kinases. The site, however, does not correspond to the active site, but to a surface-exposed loop. In the kinases and in one of the other four sites, a yeast aldose 1-epimerase, there is a TYR residue within or adjacent to the microenvironment. One or two other sulfur-containing sidechains are also present. Since the kinases are all known to be phosphorylated, it is possible that the TYR in the microenvironment may represent a phosphorylation site. In fact, TYR416 in 1K9A:A is annotated in Swiss-Prot (ID: P32577) as a putative autophosphorylation site. The other kinase-associated sites are not annotated, but it is conceivable that they may also be phosphorylation sites, perhaps by autophosphorylation. Implications for the other four sites are unclear.

Lastly, we present Clust36-Sub127, a set of five surface-exposed microenvironments (see Figure 5.15). In four out of the five cases, the CYS is accompanied by an ASP and a LYS, potentially forming a triad. Whether or not this microenvironment performs a catalytic function is unknown, but since all of these residues are known to participate in chemical reactions, it is possible that it has an active role.

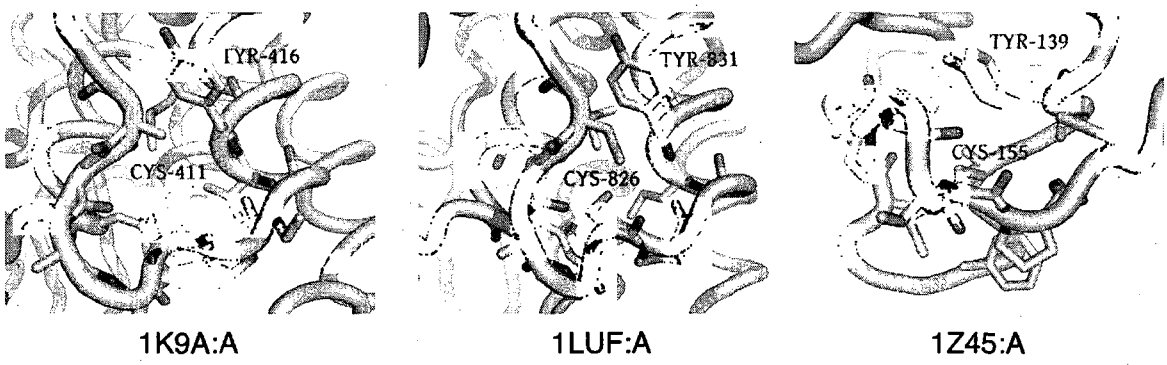


Figure 5.14: Clust5-Sub70 – A potential TYR phosphorylation site. In Clust5-Sub70, the central CYS (labeled and shown in green) is on a surface-exposed loop. Eight of the 12 microenvironments are from TYR kinases, and these microenvironments also contain a TYR residue. One of these TYR residues is annotated as a putative autophosphorylation site (TYR416 in 1K9A:A). The other seven kinase-associated microenvironments are not annotated; 1LUF:A is an example. Of the other four environments, only one – 1Z45:A, a yeast aldose 1-epimerase – contains a TYR.

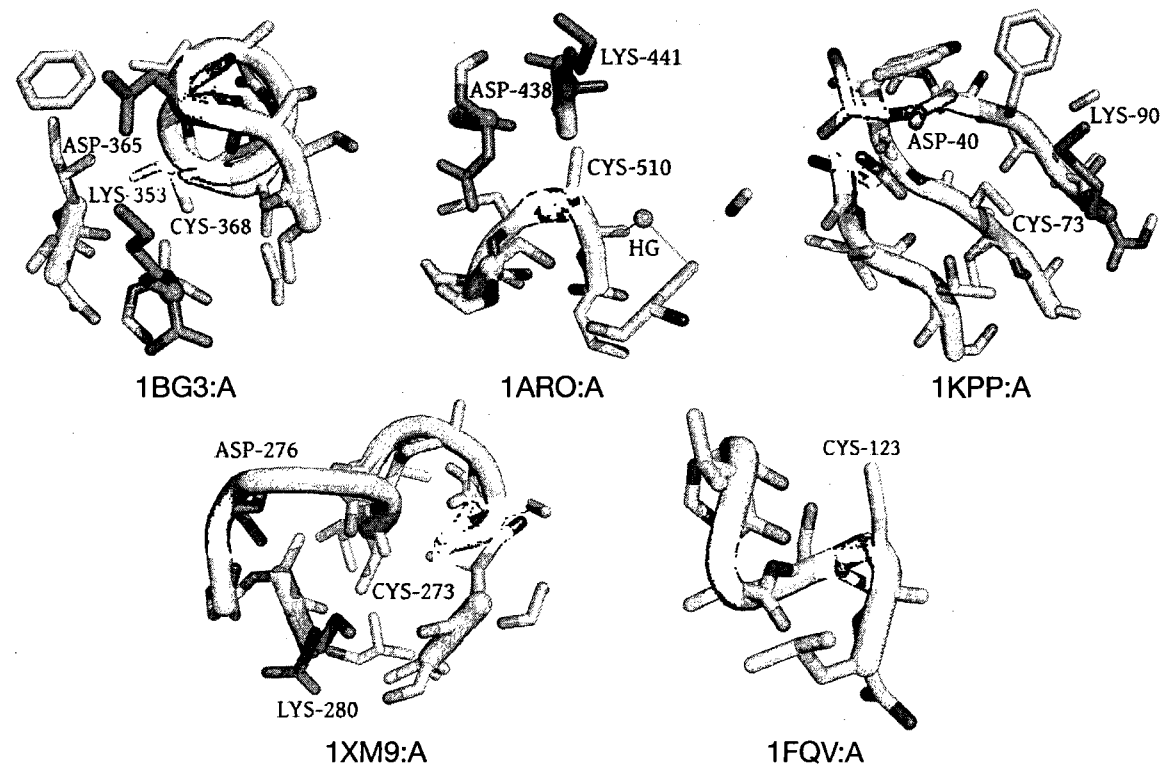


Figure 5.15: Clust36-Sub127 – A putative functional triad. In Clust36-Sub127, the central CYS (yellow) is accompanied by an ASP (red) and a LYS (blue) in four out of five cases. All of the microenvironments are surface-exposed. As CYS, ASP, and LYS are all known to be biochemically reactive, this microenvironment may have an active functional role.

5.4 Discussion

Protein function prediction has traditionally concentrated on modeling known functional domains and motifs. But just as genomics has rapidly increased the numbers of known proteins, so should it increase the number of known functions. Yet our ability to annotate new proteins lags significantly behind our ability to sequence them, and our ability to identify novel functions is almost nonexistent. In this chapter, I have presented a suite of methods that can be combined into a pipeline to analyze the results of unsupervised clustering and produce compelling, functionally characterized sub-clusters.

5.4.1 A generalizable tool for protein cluster annotation

Cluster analysis in biology reached a fever pitch several years ago, spurred by the popularization of gene expression studies. The intuitive idea of grouping together biological entities based on common features such as expression values led to a need for tools to interpret the significance of the resulting groups. It is, therefore, unsurprising that most of the tools currently available for cluster analysis are not well suited for protein microenvironment clusters spanning multiple species. With metagenomics data and aggregated data sets becoming more abundant, it is conceivable that more applications in the future will be similar to ours. A tool that can help characterize an arbitrary list of proteins is thus very useful.

With the methods I have developed and adapted, we are able to incorporate information from the PDB, Swiss-Prot, and PubMed to extract enriched terms for a cluster of protein microenvironments. Using internal coherence measures, we can evaluate how physically similar the microenvironments are, while external measures

such as functional coherence allow us to assess the amount of knowledge available to elucidate the underlying biological details. By presenting the extracted terms in a summary page, a clear functional signal is immediately obvious when it is present. In less clear cases, more in-depth exploration becomes straightforward through links to detailed term lists and outside databases. Although we applied the annotation system to a data set of FEATURE-derived microenvironments, the methods could also be applied to any list of proteins named as PDB or Swiss-Prot IDs with only minor modifications.

Despite these advantages, the methods presented here can be significantly extended. A simple improvement to our methods would be to use hierarchy information for GO and MeSH terms so that different levels of granularity are not penalized as being unique terms. Better filtering of non-functional terms would also improve the biological signal of annotations. For analyzing microenvironment-type clusters specifically, another improvement would be to provide a visualization or description of the feature vectors and a built-in protein structure viewer. This capability may depend on choices made for k-means clustering; for example, dimensions in PCA do not directly correspond to intuitive physicochemical features as in traditional FEATURE vectors, and so the biological importance of significant features will not be obvious. It is also important to note that the field of biological text mining is advancing rapidly, and many powerful techniques are available for summarizing information and uncovering relationships between biological entities and concepts. A more sophisticated annotation system might make use of named entity recognition, concept recognition, and relationship extraction to derive more robust or complex associations between cluster members.

Without some way of interpreting biological clusters, the results of unsupervised

approaches are of little use. A generalizable annotation system that draws from diverse sources of information, such as the one described here, is an effective way to summarize any coherent biological signals present in a cluster and acts as a starting point for further investigation.

5.4.2 Prioritizing clusters in unsupervised approaches for functional site discovery

One of the drawbacks of k-means clustering is that we must specify the number of clusters beforehand. Although heuristics can provide reasonable estimates, it is still a challenge to set parameters when the true structure of the data is not known. Other methods such as mixture modeling may be better suited for identifying underlying patterns but are computationally more expensive and also require parameter estimation. In this work, we demonstrate a two-step approach that would allow for fewer assumptions in the initial clustering and provide better separation in subsequent analyses. The ability to post-process large clusters into smaller, more coherent subclusters means that we do not have to attempt to divide all the objects into the most optimal groups at the outset, but can simply group them into coarse "ballparks". We can then use more accurate but more expensive methods such as hierarchical clustering to identify finer-grained distinctions within the large groups.

Additional parameter choices are necessary, and these are not always intuitive. For instance, single linkage performed better than average and complete linkage in producing high purity sub-clusters, even though the opposite has been shown [52]. It is important to note, however, that different linkage methods make fundamentally different assumptions about the underlying structure of the data. Average and complete

linkage assume spherical, well-separated clusters, while single linkage maximizes connectivity between neighboring points and is best suited for chain-like clusters. Since proteins and their functional sites have evolved over time from common ancestors, it actually makes some intuitive sense for similar microenvironments to possess a linear, chain-like relationship.

A number of different validation measures can also be used to prevent bias towards artificially small or large clusters as can occur when optimizing cluster purity or inverse purity, respectively. Internal measures can be biased towards certain types of clusters – the silhouette width favors compact and spatially separated clusters, for example. We chose cluster purity as the measure of interest for sub-cluster selection, mainly because we are interested in fine distinctions between microenviroments and do not view higher level redundancy such as multiple zinc binding site sub-clusters as a drawback. We also stipulated that sub-clusters have a minimum of five sites, which reduces some of the bias towards extremely small clusters.

In addition, we use two conceptually different evaluation measures – functional coherence and internal node correlation – to choose sub-clusters, providing further balance between potential biases. Higher node correlation favors microenvironments that are physically similar, while higher functional coherence moves towards existing evidence and knowledge. Here, we use a scoring function that is fairly equal in its weighting of these two coherence measures, but the function can be easily modified to suit particular needs. To recapture only well-known functional sites, a function heavily weighted towards functional coherence would perform better. Weighting the function more towards internal coherence would produce sub-clusters that are very physically tight, but may not have clear or meaningful biological significance.

In essence, we can modify the scoring function to make different types of discoveries. Sub-clusters that are already well characterized (as suggested by high functional coherence) may have one or two members that are not annotated with that particular function; we can then transfer the annotation indicated by the sub-cluster analysis to those members. Sub-clusters that emphasize internal coherence but have low functional coherence, on the other hand, may represent completely novel functional sites. Somewhere in between lies a third type of discovery – that of a 3-dimensional motif for a characterized function that did not previously have a defined motif. These three types of discoveries could be described as "individual protein annotation", "motif identification", and "novel functional site discovery", and each one is more difficult to validate than the former. Each type is, however, also more interesting from a scientific standpoint than the preceding type.

As reviewed in Handl *et al.* [60], cluster analysis requires considerable care in the selection of parameters, algorithms, and validation techniques due to many potential sources of bias. Because the underlying data structure is unknown, it is useful to evaluate several different classes of methods, *e.g.* a method that assumes compact clusters versus one that assumes connected clusters. Our use of k-means provides very rough spherical estimates which we then refine based on connectivity with single linkage hierarchical clustering, but evaluating different methods for each phase may help improve performance, or even replace the two phases with one if the method is suitably accurate and computationally tractable. PW-k-means [153], which incorporates a weighting and penalization scheme to incorporate prior information and reduce the damaging effect of noise points, may be a promising method to investigate. We currently do this in a discrete fashion – using functional coherence to provide prior information and the overall sub-cluster selection process to prune out singletons – but

PW-k-means covers similar aims in one integrated method and may produce good results on our data set. As I will discuss below, however, hierarchical clustering is still useful for exploratory reasons, and the discrete steps may make fine-tuning different aspects of the pipeline easier.

Our results indicate that we can identify biologically meaningful clusters of protein microenvironments using a two-step clustering and prioritization approach along with text-mining-based annotation methods. We demonstrate this by rediscovering known functional sites such as the active site for TYR phosphatases and binding sites for zinc and copper. More interestingly, we can distinguish between sub-classes of functional sites, such as the blue copper and multicopper oxidase sub-clusters, and different modes for zinc binding. In addition, some of these known functional site clusters yield potentially novel individual protein annotations which would be interesting to validate experimentally. We also present several examples of putative novel functional sites; the interpretation of such sub-clusters is challenging, but merits follow-up. Annotations from other motif databases such as Pfam, PROSITE, and Gene3D do not shed light on the putative relationship between these recurring microenvironments. Importantly, the majority of potentially novel sub-clusters contain residues that are not contiguous in sequence, but are separated by 50 or more amino acids. Traditional sequence- and structure-based alignment algorithms do not handle large gaps in the sequence or structure backbone, so they would not be able to detect recurring regions in proteins such as these.

5.4.3 Enabling exploration of protein function space

Beyond recapitulating known functions and identifying potentially novel sites, our cluster analysis approach allows open-ended exploration of protein function space as

described by microenvironments. The hierarchical trees that form the basis of the cluster selection process have inherent value; we can use them to see how similar functional microenvironments are to each other and ask interesting questions. In our 15 SeqFEATURE model test set, for example, a significant number of sites did not map to the groupings we would expect, either because their inclusion negatively impacted the internal or external coherence, or because they were located in a different area of the tree. Both cases suggest that the microenvironments for these "singleton" sites differ in some way from that of the other sites mapped to the same PROSITE pattern. What makes these sites so different, and what implications does this have for their classification? What might this say about the evolution of a particular function?

Inspection of the overall tree of sub-clusters can also lead to interesting questions, for we can see how the microenvironments of different functions relate to one another. If we again consider the 15 SeqFEATURE test set, we see that zinc protease active sites are similar to those of other zinc-containing enzymes, beta lactamases and RNases. These group with other hydrolases like alcohol dehydrogenase, serine protease (trypsin-type), and carboxylesterase. Note that the two proteases are less similar to each other than to other types of enzymes. In addition, thiol proteases are far removed in this tree and not very similar to other enzymes. These observations make sense given the diverse origins of proteases, many of which arose independently even while sharing very similar catalytic mechanisms [112, 11]. Other dissimilarities – or similarities – between different classes of enzymes may be less well known and worth investigating. It may also be possible to use hierarchical trees of microenvironments to inform protein engineering applications based on the similarity of functional sites to one another.

When we examine the results of cluster selection and annotation on unknown

data, we can ask more directed questions, such as "Does this protein bind copper?" or "Does this group of proteins share a protein-binding-related microenvironment?" Some of these questions will be easier to answer than others, but, as mentioned previously, the more challenging cases are also the most interesting. While our text-based analysis provides some initial clues, more sophisticated text mining methods and microenvironment visualizations will improve our ability to make testable hypotheses.

5.4.4 Building a pipeline for functional site discovery

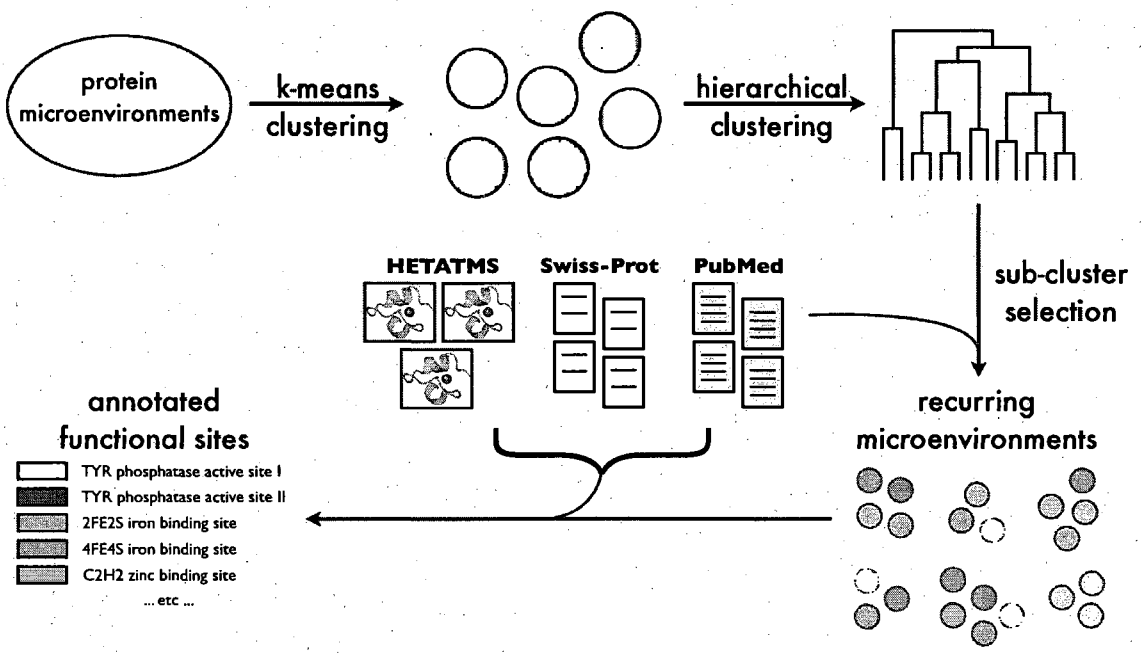


Figure 5.16: A pipeline for protein functional site discovery. By integrating different unsupervised clustering methods with existing knowledge, we are able to group protein microenvironments into coarse clusters, refine them into more relevant sub-clusters, and annotate them with useful information from curated protein databases and scientific literature.

With this work, we have demonstrated the feasibility of key components in a pipeline for discovering novel functional sites in protein structures (see Figure 5.16). Given unsupervised clustering of protein microenvironments, we can now refine and

prioritize the resulting large clusters into smaller, more compelling sub-clusters using a combination of finer-grained hierarchical clustering and scoring functions that balance internal and external coherence. We can also integrate knowledge from literature and other databases to form a picture of the underlying biological features salient in each sub-cluster. This procedure as a whole thus represents a semi-automated pipeline that will enable the prediction of novel annotations for individual proteins, 3D motifs for known functions, and potentially novel functional sites on a scale not previously feasible.

Chapter 6

Conclusions and future directions

6.1 Conclusions

The field of protein function prediction faces many challenges due to advancing technology, producing a need for robust, comprehensive, and efficient methods for recognizing potential functions in protein structures bearing little resemblance to known proteins. There is also a unique problem in that no pipeline exists for discovering, characterizing, and subsequently modeling novel functions, a need that becomes increasingly likely given the rate at which new proteins are being identified. In this dissertation, I present work on characterizing and annotating protein function using automated computational methods. The methods I developed build on many existing approaches and techniques, including natural language processing, the neighbor divergence per gene algorithm, and applications of the FEATURE framework for structure-based functional site modeling, and I have integrated them in novel ways.

6.1.1 The SeqFEATURE library

With the creation of the library of SeqFEATURE models, I contribute a comprehensive, validated tool for protein function prediction that is structure-based without over-reliance on fold or residue conservation. The SeqFEATURE approach allows the construction of large sets of functional site models quickly and automatically from sequence motifs. In our application, we show that although existing sequence- and structure-based methods have better performance in general, SeqFEATURE models perform better when sequence and structural similarity to known proteins is low. Since this scenario is typical of many structural genomics targets, SeqFEATURE should be useful for functional characterization of these structures. In addition, SeqFEATURE often correctly classifies cases that are incorrectly classified by other methods, suggesting that it is useful to include it in functional analyses of new proteins. We present several examples where our predictions on structural genomics targets support those of other methods, and also an example where we generate a novel prediction. The library of models and an interface for easy scanning of structures are available via WebFEATURE, where users can also download full data from our functional scan of the PDB.

6.1.2 Discovering novel functional sites

To address the problem of discovering and characterizing novel functions, I employed techniques from natural language processing and cluster analysis to develop cluster selection and annotation methods. We use the neighbor divergence per gene algorithm to assess a cluster's functional coherence. Functional clusters derived from PROSITE achieve a much higher functional coherence than random clusters of similar sizes, and functional coherence degrades as functional signal decreases, indicating that this

measure is a suitable proxy for functional signal. We combine functional coherence with an internal coherence metric into a scoring function, which we use in concert with hierarchical clustering to select optimal sub-clusters within large, k-means-derived clusters.

Once we have obtained candidate sub-clusters, we use an annotation system I developed that incorporates information from literature and other databases to aid in characterization. Terms from PubMed abstracts, Swiss-Prot records, and PDB HETATMs receive scores based on the hypergeometric distribution; terms from PubMed abstracts are additionally scored for relevance using a novel entropy-based scoring function. Comparing terms from controlled vocabularies such as MeSH to raw text demonstrates the tradeoffs between the two types of data; although MeSH terms are often more conceptually clear, they can be less specific and informative than terms extracted from raw text. The inclusion of Swiss-Prot data and HETATMs provides additional facets for analysis. When we evaluate literature and GO-based term lists on clusters with decreasing functional signal, we see a sharp drop-off in performance below about 40% signal, underscoring the need to produce reasonably coherent clusters to begin with. We present all of the information produced by the annotation method in a summary page that is hyperlinked to more detailed pages as well as external databases.

Applying these methods to a data set built from CYS-centered FEATURE microenvironments yielded promising and interesting results. We rediscovered known sites, such as TYR phosphatase active sites and several metal-binding sites, predicted novel zinc-binding annotations for individual proteins, and presented a novel functional sites potentially related to structural stability, TYR phosphorylation, and catalytic activity.

6.2 Contributions to informatics

In this dissertation, I explored supervised and unsupervised techniques for protein function characterization. With SeqFEATURE, I demonstrated that 3D models built using the FEATURE framework are more robust than methods that rely solely on sequence and structure conservation, suggesting that this approach will be useful for characterizing novel protein structures. The pipeline as a whole illustrates a method by which a large library of 3D functional site models can be constructed automatically from a set of 1D sequence motifs.

In addition, I have developed methods that can be used in conjunction with large-scale unsupervised clustering studies. These methods allow the refinement and selection of compelling sub-clusters within larger, coarse-grained clusters, and the subsequent characterization of these sub-clusters using information from external knowledge sources. The cluster selection process, when applied to known functional sites, also encourages exploration of protein function space using a flexible, discontinuous representation, inspiring interesting questions about functional site relatedness, evolution, and engineering. These methods taken together represent a framework for discovering and characterizing potentially novel functional sites in protein structures.

6.3 Contributions to biomedicine

Through this dissertation work, I have created a large, validated library of 3D functional site models which can be used to scan protein structures for function. We have used the library to scan the entire PDB and have made predictions of function for unannotated structural genomics targets. I have also applied the sub-cluster selection and characterization methods to a novel data set, recapitulating many known

functional sites as well as uncovering intriguing, potentially novel discoveries. Unsupervised clustering combined with external knowledge sources allows the definition of biologically relevant sub-classes of functional sites, such as different modes of zinc binding or functional distinctions between copper-binding proteins, perhaps indicating that microenvironments are a useful way to explore and compartmentalize protein structure and function space.

6.4 Future directions

6.4.1 Modeling of known functions

As mentioned above, these methods are a promising starting point for more comprehensive studies and modeling of protein functional sites. A straightforward extension of SeqFEATURE would be to apply it to additional motif databases. More accurate models are possible if training set construction takes into account false negatives and false positives for each motif, adding them to the positive and negative training sets, respectively. SeqFEATURE also contains multiple models for many PROSITE patterns, and the number and location of hits to these models can be informative. A compound model approach where the results for multiple models are considered could reduce false positives.

There are many areas where FEATURE itself could also be improved. Its strengths are its robustness and descriptive microenvironment representation, but there is much more information available than FEATURE takes into account. For instance, we could determine from multiple sequence alignments which residues in a functional site are conserved and which can be mutated; this can result in better detection of functional sites which have either converged from different evolutionary origins, or

diverged in areas of the protein that are not necessary for function. FEATURE's microenvironment vectors are easily amenable to the addition of new properties.

Additional improvements include consideration of site geometry and better modeling of features as a function of distance from the site center. Although spherically averaged models are computationally inexpensive and statistically robust, the relative location of atoms or residues from one another can often be crucially important for function. One could imagine capturing orientation data indirectly, perhaps by computing distances between pairs of sidechains, without significantly increasing computational cost. To represent microenvironments even more robustly, we could describe feature distributions within and across shells using a continuous, statistical model. In this way, outliers and empty values in the training set could be smoothed, and deviations from the norm would be weighted according to the learned model.

Because information about protein function can be detected and encapsulated in so many ways, it is unlikely that any one method will outperform all others in every scenario. In fact, many hybrid techniques exist that apply multiple methods to improve functional coverage and increase the chances of producing high confidence predictions. We have begun to do this with FEATURE in a way, by coupling the use of FEATURE models with molecular simulations to detect function in situations where FEATURE by itself falls short. A more conventional approach, however, would be to integrate methods like the ones we used in our comparison with SeqFEATURE, turning the outputs from multiple *in silico* assays into inputs for an overall classifier.

6.4.2 Cluster analysis and novel site discovery

There are many ways in which textual knowledge can be leveraged to aid in cluster analysis. Although neighbor divergence per gene is reasonably effective, it may

be helpful to employ more sophisticated methods for determining semantic similarity between documents. Likewise, advanced NLP techniques such as named entity recognition and relationship extraction may provide more useful literature-based annotations for clusters. The information incorporated into the annotation system can also be expanded to include knowledge from pathways, hierarchies of controlled vocabularies, and homologous proteins, to name only a few.

On its own, the literature-based annotation approaches may have additional utility. Neighbor divergence per gene and multi-faceted term enrichment annotation are relatively straightforward and likely to be widely applicable; I demonstrated their application to arbitrary lists of proteins. With large amounts of data now available on genes, drugs, proteins, and other interesting biological agents, methods that can help determine the significance of particular groupings – structurally similar drugs, for example – could prove to have great impact.

In the case of FEATURE microenvironment clusters, techniques to visualize the significant properties defining the microenvironments comprising particular clusters would certainly help interpretation. Currently, we transform the property vectors into principal components for clustering, and while we can easily identify which components differ between sub-clusters, it is difficult to translate this into an intuitive understanding of the protein microenvironments. Converting principal components of interest back into their original physicochemical properties and highlighting the differences visually – perhaps even in the structures themselves – would be useful for gaining fuller comprehension of potentially subtle differences.

As discussed in Section 5.4.3, our unsupervised clustering approaches such as hierarchical clustering encourage exploration and hypothesis generation even with well-characterized data sets. Identification and elucidation of surprising relationships and

memberships (or the lack thereof) may help improve our understanding of the association between protein structure and function as well as protein evolution. Currently, our best protein classification schemes incorporate information based on sequence and structure homology mainly at the semi-global fold level. We could imagine, however, a microenvironment-based classification that considers local, discontinuous regions and only indirectly considers the evolutionary relationship between protein structures. Such a classification may be especially helpful for understanding protein structure and function from an engineering perspective. Exciting research, both discovery-focused and descriptive in nature, is clearly possible from the outputs of unsupervised learning on protein microenvironments.

Once additional clustering data sets are available, the cluster selection and annotation methods can be easily applied to produce even more interesting findings and work towards a comprehensive description of protein microenvironment space. With enough supporting evidence, clusters also logically serve as training sets for supervised modeling of the newly discovered or rediscovered functions, making a complete, semi-automated pipeline for protein functional site discovery and modeling possible.

Appendix A

SeqFEATURE supplementary data

This section contains performance statistics for the SeqFEATURE library of models, and the data sets required to reproduce the library and carry out the comparison against other methods. Note that only positive training sets are listed, as the negative training sets are generated randomly as described in Section 3.1.1. Performance statistics can also be viewed on WebFEATURE by selecting the model in the drop-down menu and clicking on "more info."

A.1 SeqFEATURE model performance

See Table A.1 for a list of performance statistics for the entire library of SeqFEATURE models. For each model, we measured the area under the ROC curve (AUC) and the sensitivity (based on training sets) at each of three specificity-based score cutoffs (100% specificity, or '100c'; 99% specificity, or '99c'; and 95% specificity, or '95c'). All score cutoffs are shown as Z-scores, normalized to the overall distribution of scores for the corresponding training set.

Table A.1: Performance statistics for SeqFEATURE models.

Model name	AUC	100c	100-sens	99c	99-sens	95c	95-sens
2FE2S_FERREDOXIN.1.CYS.SG	0.9600	4.1959	0.7778	2.3834	0.8889	1.8354	0.8889
2FE2S_FERREDOXIN.6.CYS.SG	0.8992	5.5715	0.0000	2.5647	0.7778	1.6712	0.7778
2FE2S_FERREDOXIN.9.CYS.SG	0.8544	4.9675	0.2222	2.6517	0.7778	1.6671	0.7778
4FE4S_FERREDOXIN.1.CYS.SG	0.9998	3.1084	0.1000	2.0501	1.0000	1.5589	1.0000
4FE4S_FERREDOXIN.3.CYS.SG	0.9991	3.4191	0.0000	2.0044	0.9500	1.4919	1.0000
4FE4S_FERREDOXIN.5.CYS.SG	0.9999	2.7500	0.4000	1.5623	1.0000	1.2660	1.0000
4FE4S_FERREDOXIN.7.CYS.SG	0.9054	5.2182	0.2500	2.4541	0.9500	1.5971	0.9500
AA_TRANSFER_CLASS_1.4.LYS.NZ	0.9573	3.8958	0.1667	2.6897	0.8333	1.8763	0.8333
AA_TRANSFER_CLASS_2.4.LYS.NZ	0.5215	5.0026	0.0000	2.9560	0.0000	1.9219	0.4000
AA_TRANSFER_CLASS_3.19.LYS.NZ	0.9931	4.6946	0.0000	2.8073	0.8000	2.0864	1.0000
ADH_SHORT.3.TYR.OH	0.9999	5.0745	0.1176	2.2891	1.0000	1.6249	1.0000
ADH_ZINC.2.HIS.ND1	1.0000	3.4172	1.0000	2.4184	1.0000	1.7128	1.0000
ADH_ZINC.2.HIS.NE2	0.9996	3.8970	0.6667	2.4840	1.0000	1.7499	1.0000
ADX.6.CYS.SG	0.9357	5.1797	0.6667	2.8744	0.8333	1.8342	0.8333
ADX.9.CYS.SG	0.8905	6.6588	0.6667	2.8548	0.8333	1.7823	0.8333
ALDEHYDE_DEHYDR_CYS.6.CYS.SG	0.2667	4.6874	0.0000	2.6893	0.0000	1.5310	0.0000
ALDEHYDE_DEHYDR_GLU.2.GLU.OE1	0.3238	5.0167	0.0000	2.7511	0.0000	1.8296	0.4000
ALDEHYDE_DEHYDR_GLU.2.GLU.OE2	0.4793	4.7775	0.0000	2.6697	0.0000	1.8642	0.2000
ASP_PROTEASE.4.ASP.OD1	0.9964	4.6587	0.0588	2.3194	0.8824	1.7303	1.0000
ASP_PROTEASE.4.ASP.OD2	0.9994	3.7837	0.4706	2.2973	1.0000	1.7238	1.0000
ASX_HYDROXYL.3.ASN.ND2	0.9856	4.6894	0.2000	2.8130	0.6000	1.8640	1.0000
ASX_HYDROXYL.3.ASN.OD1	0.9681	5.0918	0.2000	2.6115	0.8000	1.8262	0.8000
BETA_LACTAMASE_A.5.SER.OG	0.9983	4.0089	0.8000	2.4755	1.0000	1.7438	1.0000
BETA_LACTAMASE_B_1.4.HIS.ND1	0.9993	3.8683	0.8000	2.6032	1.0000	1.8239	1.0000
BETA_LACTAMASE_B_1.4.HIS.NE2	0.9902	4.8795	0.8000	2.8145	0.8000	1.7450	1.0000
BETA_LACTAMASE_B_1.6.HIS.ND1	0.9997	5.3466	0.8000	2.7205	1.0000	1.7821	1.0000
BETA_LACTAMASE_B_1.6.HIS.NE2	0.9949	5.5322	0.0000	2.8292	0.6000	1.8709	1.0000
BETA_LACTAMASE_B_1.8.ASP.OD1	0.9991	4.6363	0.6000	2.9202	1.0000	1.8558	1.0000
BETA_LACTAMASE_B_1.8.ASP.OD2	0.9982	5.4960	0.6000	3.0055	1.0000	1.9485	1.0000
BPTI_KUNITZ_1.4.CYS.SG	0.9943	2.8687	0.1667	2.3079	0.6667	1.7660	1.0000
BPTI_KUNITZ_1.8.CYS.SG	0.9999	3.5059	0.8333	2.2843	1.0000	1.7820	1.0000
CARBOXYLESTERASE_B_1.11.SER.OG	1.0000	5.2823	1.0000	2.5415	1.0000	1.7047	1.0000
CARBOXYLESTERASE_B_2.3.CYS.SG	0.9837	4.6941	0.6667	2.6651	0.8333	1.8180	0.8333
CHITINASE_18.9.GLU.OE1	0.8890	5.8069	0.2000	2.9593	0.6000	1.9035	0.6000
CHITINASE_18.9.GLU.OE2	0.8423	4.0715	0.6000	2.7903	0.8000	1.8832	0.8000
COPPER_BLUE.11.HIS.ND1	0.8889	4.4932	0.7273	2.3960	0.9091	1.7011	0.9091
COPPER_BLUE.11.HIS.NE2	0.9144	3.2990	0.5455	2.0883	0.9091	1.5094	0.9091

COPPER_BLUE.7.CYS.SG	0.9976	4.9485	0.5455	2.5586	0.9091	1.7391	1.0000
CYTOCHROME_P450.8.CYS.SG	1.0000	4.1254	0.8333	2.4019	1.0000	1.7526	1.0000
C_TYPE_LECTIN_1.1.CYS.SG	0.9759	3.1375	0.1667	1.9043	0.9167	1.6408	0.9167
EF_HAND.1.ASP.OD1	0.8853	5.9361	0.2698	2.5044	0.8571	1.7609	0.8730
EF_HAND.1.ASP.OD2	0.8666	4.2962	0.3968	2.3821	0.8254	1.7175	0.8413
EF_HAND.12.TYR.OH	0.9836	4.4758	0.0000	2.9857	0.6000	1.9715	1.0000
EF_HAND.3.ASN.ND2	0.7622	3.9446	0.0000	1.8989	0.5417	1.3919	0.7083
EF_HAND.3.ASN.OD1	0.8451	5.8858	0.1667	2.2276	0.7083	1.4801	0.7917
EF_HAND.3.ASP.OD1	0.8558	4.9395	0.4706	2.4333	0.8529	1.5919	0.8824
EF_HAND.3.ASP.OD2	0.9664	3.6033	0.0294	1.9822	0.8529	1.5595	0.8824
EF_HAND.5.ASN.ND2	0.6417	4.1280	0.0000	2.0562	0.6000	1.5011	0.6000
EF_HAND.5.ASN.OD1	0.5287	5.0505	0.4000	2.6283	0.6000	1.7555	0.6000
EF_HAND.5.ASP.OD1	0.9096	4.4184	0.3784	2.4991	0.8378	1.7351	0.9189
EF_HAND.5.ASP.OD2	0.8905	3.3269	0.1351	2.0530	0.7297	1.5672	0.8919
EF_HAND.5.SER.OG	0.5918	4.2569	0.4167	2.3425	0.5833	1.6453	0.6667
EF_HAND.7.GLU.OE1	0.6573	4.2014	0.0000	2.6910	0.1429	1.9192	0.4286
EF_HAND.7.GLU.OE2	0.7653	4.1792	0.0000	2.4665	0.1429	1.8074	0.2857
EF_HAND.7.LYS.NZ	0.1233	5.5667	0.0000	2.4721	0.0000	1.9064	0.2000
EF_HAND.7.THR.OG1	0.9687	3.6176	0.4545	2.4299	0.8182	1.6725	0.9091
EF_HAND.7.TYR.OH	0.9006	2.4005	0.0000	2.1812	0.0000	1.9558	0.5000
EF_HAND.9.ASN.ND2	0.7788	4.1066	0.0000	3.0109	0.3333	2.0911	0.6667
EF_HAND.9.ASN.OD1	0.8631	4.9062	0.1667	2.6925	0.6667	1.8499	0.6667
EF_HAND.9.ASP.OD1	0.7061	6.8989	0.1875	2.8871	0.6875	1.6765	0.7500
EF_HAND.9.ASP.OD2	0.4321	3.8412	0.0625	2.2793	0.2500	1.5232	0.4375
EF_HAND.9.SER.OG	0.8184	4.4551	0.0000	3.0052	0.5238	1.8453	0.8095
EF_HAND.9.THR.OG1	0.9198	4.7841	0.0000	2.9668	0.4000	1.8192	0.6000
EGF_1.1.CYS.SG	0.9433	2.7891	0.3600	2.0501	0.7600	1.6619	0.8000
EGF_1.3.CYS.SG	0.9724	2.6989	0.3333	2.0497	0.8750	1.7102	0.8750
EGF_1.7.CYS.SG	0.9531	2.9954	0.2609	2.1801	0.7826	1.7073	0.8696
EGF_2.1.CYS.SG	0.7922	2.1644	0.1000	1.8328	0.4000	1.6361	0.5000
EGF_2.3.CYS.SG	0.9186	2.6325	0.1111	2.0993	0.4444	1.7178	0.8889
EGF_2.8.CYS.SG	0.7934	2.6486	0.0000	2.0233	0.2222	1.7165	0.4444
GLYCOSYL_HYDROL_F10.7.GLU.OE1	0.7618	3.8358	0.6667	2.5048	0.6667	1.8222	0.6667
GLYCOSYL_HYDROL_F10.7.GLU.OE2	0.7614	4.0324	0.6667	2.6091	0.6667	1.9083	0.6667
GLYCOSYL_HYDROL_F5.7.GLU.OE1	1.0000	3.8385	1.0000	2.7188	1.0000	1.8787	1.0000
GLYCOSYL_HYDROL_F5.7.GLU.OE2	0.9998	3.9203	0.8333	2.6280	1.0000	1.8757	1.0000
HIPIP.1.CYS.SG	0.8511	5.5838	0.6000	3.0376	0.8000	1.8222	0.8000
HIPIP.7.CYS.SG	1.0000	4.8473	1.0000	2.7401	1.0000	1.7795	1.0000
HMA_1.5.CYS.SG	0.9802	3.8045	0.0000	3.0551	0.6667	2.0610	0.8889
HMA_1.8.CYS.SG	0.9590	3.0503	0.0000	2.2761	0.7778	1.7066	0.7778

IG_MHC.3.CYS.SG	0.9009	4.5937	0.0000	2.6080	0.8000	1.6692	0.8000
IMP_1.4.ASP.OD1	0.9569	4.3603	0.6000	2.5989	0.8000	1.8058	0.8000
IMP_1.4.ASP.OD2	0.9994	3.9608	0.6000	2.5508	1.0000	1.8129	1.0000
KAZAL.1.CYS.SG	0.9222	2.0286	0.2000	1.6858	0.4000	1.5563	0.6000
KAZAL.3.CYS.SG	0.9732	2.1815	0.2000	1.8104	0.8000	1.6138	0.8000
KAZAL.7.CYS.SG	0.9704	2.2009	0.2000	1.9541	0.8000	1.6396	0.8000
KAZAL.9.CYS.SG	0.9752	2.0881	0.0000	1.8642	0.4000	1.6227	1.0000
LIPASE_SER.7.SER.OG	0.9995	4.7293	0.6250	2.5387	1.0000	1.7350	1.0000
LIPOYL.9.LYS.NZ	0.3463	5.7718	0.0000	2.7817	0.1429	1.7781	0.2857
PA2_HIS.4.HIS.ND1	0.7397	3.8229	0.6000	1.9125	0.6000	1.4144	0.6000
PA2_HIS.4.HIS.NE2	0.6019	4.8345	0.6000	2.7310	0.8000	1.7391	0.8000
PEROXIDASE_1.8.HIS.ND1	0.7266	3.8764	0.2857	2.3347	0.7143	1.6333	0.7143
PEROXIDASE_1.8.HIS.NE2	0.5890	5.0036	0.0000	2.5812	0.1429	1.7305	0.4286
PEROXIDASE_2.8.HIS.ND1	0.9999	3.6628	0.8000	2.6225	1.0000	1.7515	1.0000
PEROXIDASE_2.8.HIS.NE2	0.9997	3.9233	0.6000	2.5753	1.0000	1.8144	1.0000
PHOSPHOPANTETHEINE.6.SER.OG	0.0012	5.3856	0.0000	3.4820	0.0000	1.9401	0.0000
PROTEIN_KINASE_ST.5.ASP.OD1	0.9456	3.6536	0.6500	2.5218	0.8500	1.8006	0.9000
PROTEIN_KINASE_ST.5.ASP.OD2	0.9700	3.8913	0.5000	2.5213	0.8000	1.7986	0.9500
PTS_HPR_SER.5.SER.OG	0.4289	5.5779	0.0000	2.4075	0.1667	1.5498	0.1667
RNASE_T2_1.4.HIS.ND1	0.9651	4.0333	0.2000	2.5135	0.6000	1.8737	0.8000
RNASE_T2_1.4.HIS.NE2	1.0000	4.0910	1.0000	2.6209	1.0000	1.6521	1.0000
SHIGA_RICIN.5.GLU.OE1	0.9959	3.8771	0.5714	2.5391	0.8571	1.8446	1.0000
SHIGA_RICIN.5.GLU.OE2	0.9893	3.4547	0.7143	2.5786	0.8571	1.9350	0.8571
SHIGA_RICIN.8.ARG.NE	0.9811	4.6423	0.4286	2.3878	0.5714	1.7262	1.0000
SHIGA_RICIN.8.ARG.NH1	0.9257	3.5732	0.1429	2.3931	0.7143	1.7380	0.8571
SHIGA_RICIN.8.ARG.NH2	0.9914	3.6414	0.1429	2.4906	0.8571	1.7443	1.0000
SMALL_CYTOKINES_CC.1.CYS.SG	0.9002	2.0993	0.0000	1.7729	0.0000	1.5961	0.3333
SMALL_CYTOKINES_CC.11.CYS.SG	0.8937	2.1243	0.0000	1.7708	0.3333	1.5981	0.3333
SMALL_CYTOKINES_CC.17.CYS.SG	0.9704	1.9338	0.0000	1.7389	0.3333	1.6214	0.8333
SMALL_CYTOKINES_CC.2.CYS.SG	0.9611	2.1115	0.0000	1.7919	0.6667	1.6081	0.8333
SNAKE_TOXIN.2.CYS.SG	0.9596	2.4282	0.5000	1.9188	0.5000	1.6438	0.8333
SNAKE_TOXIN.4.CYS.SG	0.9371	2.4193	0.0000	1.9390	0.6667	1.6970	0.8333
SNAKE_TOXIN.7.CYS.SG	0.9777	2.4263	0.0000	1.8772	0.8333	1.6479	0.8333
SNAKE_TOXIN.8.CYS.SG	0.9627	2.4496	0.8333	1.9918	0.8333	1.7157	0.8333
SUBTILASE_ASP.5.ASP.OD1	0.7117	4.4043	0.0000	2.5495	0.3333	1.7552	0.3333
SUBTILASE_ASP.5.ASP.OD2	0.7510	5.1560	0.0000	2.6061	0.3333	1.9038	0.5000
THIOL_PROTEASE_ASN.6.ASN.ND2	1.0000	3.8379	1.0000	2.5982	1.0000	1.8237	1.0000
THIOL_PROTEASE_ASN.6.ASN.OD1	1.0000	4.1252	1.0000	2.6255	1.0000	1.7985	1.0000
THIOL_PROTEASE_HIS.3.HIS.ND1	0.6356	3.9845	0.2500	2.5192	0.6250	1.7472	0.6250
THIOL_PROTEASE_HIS.3.HIS.NE2	0.6890	6.7784	0.0000	2.6849	0.6250	1.7437	0.6250

THIOREDOXIN.11.CYS.SG	0.8033	3.1060	0.4286	2.3673	0.7143	1.8022	0.7143
THIOREDOXIN.8.CYS.SG	0.7670	4.1475	0.1429	2.4014	0.7143	1.7187	0.7143
TRYPSIN_HIS.5.HIS.ND1	0.9446	6.6747	0.0588	2.4167	0.8824	1.6437	0.8824
TRYPSIN_HIS.5.HIS.NE2	0.9147	5.3687	0.0588	2.2537	0.8824	1.6162	0.8824
TRYPSIN_SER.6.SER.OG	0.9998	5.4646	0.0000	2.1696	1.0000	1.6085	1.0000
TYR_PHOSPHATASE_1.3.CYS.SG	1.0000	5.4246	1.0000	2.7473	1.0000	1.7596	1.0000
UBIQUITIN_CONJUGAT_1.10.CYS.SG	0.9929	3.2398	0.6667	2.6153	0.8333	1.7964	1.0000
ZINC_FINGER_C2H2_1.1.CYS.SG	0.9958	5.1093	0.0000	2.7908	0.9375	1.7484	0.9375
ZINC_FINGER_C2H2_1.3.CYS.SG	0.9887	4.0843	0.4375	2.4635	0.9375	1.6719	0.9375
ZINC_FINGER_C2H2_1.7.HIS.ND1	0.7011	3.8728	0.4706	2.3210	0.7647	1.6784	0.8235
ZINC_FINGER_C2H2_1.7.HIS.NE2	0.8463	6.4856	0.1176	2.4277	0.8824	1.6668	0.8824
ZINC_FINGER_C2H2_1.9.HIS.ND1	0.9141	3.7644	0.1176	1.8198	0.6471	1.3989	0.7647
ZINC_FINGER_C2H2_1.9.HIS.NE2	0.9150	5.3143	0.0000	2.3368	0.8824	1.6398	0.8824
ZINC_PROTEASE.3.HIS.ND1	0.8814	3.9136	0.3889	2.4831	0.7778	1.7811	0.8333
ZINC_PROTEASE.3.HIS.NE2	0.8720	4.8449	0.0556	2.5185	0.7778	1.7717	0.8333
ZINC_PROTEASE.4.GLU.OE1	0.8915	3.5562	0.5000	2.3600	0.8333	1.7335	0.8333
ZINC_PROTEASE.4.GLU.OE2	0.8279	4.2847	0.4444	2.5055	0.7778	1.7901	0.8333
ZINC_PROTEASE.7.HIS.ND1	0.8638	3.7015	0.0000	2.1759	0.6667	1.6324	0.7222
ZINC_PROTEASE.7.HIS.NE2	0.9115	5.4711	0.0000	2.4483	0.7778	1.6937	0.8889

A.2 Positive training sets

The table below lists the positive training sets for each SeqFEATURE model. The list is organized by PROSITE pattern; the model specification indicates the position, residue, and atom at which the corresponding model is centered. The site lists specify the PDB ID and chain ID, and the residue ID of each site used in training that particular model.

Table A.2: Positive training sets for SeqFEATURE models.

Model specification	Positive sites
	2FE2S_FERREDOXIN
1.CYS.SG	(1CZP:A, CYS41), (1DOI:_, CYS63), (1FO4:A, CYS43), (1JQ4:A, CYS42), (1KF6:B, CYS57), (1KRH:A, CYS41), (1O8R:A, CYS86), (1QLA:B, CYS57), (2PIA:_, CYS272)

6.CYS.SG	(1CZP:A, CYS46), (1DOI:_, CYS68), (1FO4:A, CYS48), (1JQ4:A, CYS47), (1KF6:B, CYS62), (1KRH:A, CYS46), (1O8R:A, CYS91), (1QLA:B, CYS62), (2PIA:_, CYS277)
9.CYS.SG	(1CZP:A, CYS49), (1DOI:_, CYS71), (1FO4:A, CYS51), (1JQ4:A, CYS50), (1KF6:B, CYS65), (1KRH:A, CYS49), (1O8R:A, CYS94), (1QLA:B, CYS65), (2PIA:_, CYS280)

4FE4S_FERREDOXIN

1.CYS.SG	(1FEH:A, CYS147), (1FEH:A, CYS190), (1GTE:A, CYS986), (1H98:A, CYS39), (1HFE:L, CYS35), (1HFE:L, CYS66), (1JB0:C, CYS10), (1JB0:C, CYS47), (1JNR:B, CYS47), (1KEK:A, CYS689), (1KEK:A, CYS745), (1KF6:B, CYS148), (1KQF:B, CYS133), (1NEK:B, CYS149), (1QLA:B, CYS151), (1VJW:_, CYS10), (1XER:_, CYS83), (2FDN:_, CYS8), (2FDN:_, CYS37), (7FD1:A, CYS39)
3.CYS.SG	(1FEH:A, CYS150), (1FEH:A, CYS193), (1GTE:A, CYS989), (1H98:A, CYS42), (1HFE:L, CYS38), (1HFE:L, CYS69), (1JB0:C, CYS13), (1JB0:C, CYS50), (1JNR:B, CYS50), (1KEK:A, CYS692), (1KEK:A, CYS748), (1KF6:B, CYS151), (1KQF:B, CYS136), (1NEK:B, CYS152), (1QLA:B, CYS154), (1VJW:_, CYS13), (1XER:_, CYS86), (2FDN:_, CYS11), (2FDN:_, CYS40), (7FD1:A, CYS42)
5.CYS.SG	(1FEH:A, CYS153), (1FEH:A, CYS196), (1GTE:A, CYS992), (1H98:A, CYS45), (1HFE:L, CYS41), (1HFE:L, CYS72), (1JB0:C, CYS16), (1JB0:C, CYS53), (1JNR:B, CYS53), (1KEK:A, CYS695), (1KEK:A, CYS751), (1KF6:B, CYS154), (1KQF:B, CYS139), (1NEK:B, CYS155), (1QLA:B, CYS157), (1VJW:_, CYS16), (1XER:_, CYS89), (2FDN:_, CYS14), (2FDN:_, CYS43), (7FD1:A, CYS45)
7.CYS.SG	(1FEH:A, CYS157), (1FEH:A, CYS200), (1GTE:A, CYS996), (1H98:A, CYS49), (1HFE:L, CYS45), (1HFE:L, CYS76), (1JB0:C, CYS20), (1JB0:C, CYS57), (1JNR:B, CYS57), (1KEK:A, CYS699), (1KEK:A, CYS755), (1KF6:B, CYS158), (1KQF:B, CYS143), (1NEK:B, CYS159), (1QLA:B, CYS161), (1VJW:_, CYS20), (1XER:_, CYS93), (2FDN:_, CYS18), (2FDN:_, CYS47), (7FD1:A, CYS49)

AA_TRANSFER_CLASS_1

4.LYS.NZ	(1AJS:A, LYS258), (1GDE:A, LYS233), (1LC5:A, LYS216), (1M7Y:A, LYS273), (1O4S:A, LYS234), (1QIS:A, LYS258)

AA_TRANSFER_CLASS_2

4.LYS.NZ	(1BS0:A, LYS236), (1DQU:A, LYS473), (1FC4:A, LYS244), (1FG7:A, LYS214), (1LSS:A, LYS17)

AA_TRANSFER_CLASS_3	
19.LYS.NZ	(1GTX:A, LYS329), (1QJ5:A, LYS274), (2DKB:_, LYS272), (2GSA:A, LYS273), (2OAT:A, LYS292)
ADH_SHORT	
3.TYR.OH	(1B16:A, TYR151), (1BDB:_, TYR155), (1CYD:A, TYR149), (1E7W:A, TYR194), (1EDO:A, TYR167), (1FMC:A, TYR159), (1GCO:A, TYR158), (1GEG:A, TYR152), (1HDC:A, TYR152), (1HXH:A, TYR151), (1JA9:A, TYR178), (1JTV:A, TYR155), (1N5D:A, TYR193), (1NXQ:A, TYR155), (1OOE:A, TYR143), (1UAY:A, TYR148), (2AE2:A, TYR159)
ADH_ZINC	
2.HIS.ND1, 2.HIS.NE2	(1E3J:A, HIS66), (1F8F:A, HIS65), (1HET:A, HIS67), (1JQB:A, HIS1059), (1JVB:A, HIS68), (1KOL:A, HIS67)
ADX	
6.CYS.SG	(1AYF:A, CYS52), (1AYF:A, CYS52), (1B9R:A, CYS45), (1E9M:A, CYS45), (1E9M:A, CYS45), (1I7H:A, CYS48)
9.CYS.SG	(1AYF:A, CYS55), (1AYF:A, CYS55), (1B9R:A, CYS48), (1E9M:A, CYS48), (1E9M:A, CYS48), (1I7H:A, CYS51)
ALDEHYDE_DEHYDR_CYS	
6.CYS.SG	(1AD3:A, CYS243), (1AMU:A, CYS376), (1EUH:A, CYS284), (1JR2:A, CYS119), (1OBZ:A, CYS166), (1QJ4:A, CYS81)
ALDEHYDE_DEHYDR_GLU	
2.GLU.OE1, 2.GLU.OE2	(1AD3:A, GLU209), (1FNA:_, GLU38), (1KQ3:A, GLU244), (1LW7:A, GLU235), (1OO4:A, GLU268)
ASP_PROTEASE	
4.ASP.OD1, 4.ASP.OD2	(1FKN:A, ASP32), (1FMB:_, ASP25), (1HRN:A, ASP32), (1HRN:A, ASP215), (1J71:A, ASP32), (1J71:A, ASP218), (1KZK:A, ASP25), (1LF2:A, ASP34), (1LF2:A, ASP214), (1MPP:_, ASP32), (1MPP:_, ASP215), (1OEW:A, ASP35), (1OEW:A, ASP218), (2APR:_, ASP35), (2APR:_, ASP218), (2RSP:A, ASP37), (4FIV:_, ASP30)
ASX_HYDROXYL	
3.ASN.ND2, 3.ASN.OD1	(1DX5:I, ASN439), (1EMO:_, ASN2144), (1HZ8:A, ASN57), (1LMJ:A, ASN22), (1NZI:A, ASN134)

BETA_LACTAMASE_A	
5.SER.OG	(1BSG:_, SER70), (1CI9:A, SER75), (1E25:A, SER70), (1GHP:A, SER70), (1M40:A, SER70)
BETA_LACTAMASE_B_1	
4.HIS.ND1, 4.HIS.NE2	(1A7T:A, HIS82), (1JJE:A, HIS77), (1M2X:A, HIS116), (1SML:A, HIS84), (2BC2:A, HIS86)
6.HIS.ND1, 6.HIS.NE2	(1A7T:A, HIS84), (1JJE:A, HIS79), (1M2X:A, HIS118), (1SML:A, HIS86), (2BC2:A, HIS88)
8.ASP.OD1, 8.ASP.OD2	(1A7T:A, ASP86), (1JJE:A, ASP81), (1M2X:A, ASP120), (1SML:A, ASP88), (2BC2:A, ASP90)
BPTI_KUNITZ_1	
4.CYS.SG	(1BIK:_, CYS59), (1BUN:B, CYS40), (1DTX:_, CYS40), (1G6X:A, CYS38), (1KTH:A, CYS38), (1SHP:_, CYS36)
8.CYS.SG	(1BIK:_, CYS72), (1BUN:B, CYS53), (1DTX:_, CYS53), (1G6X:A, CYS51), (1KTH:A, CYS51), (1SHP:_, CYS49)
CARBOXYLESTERASE_B_1	
11.SER.OG	(1DX4:A, SER238), (1EA5:A, SER200), (1LLF:A, SER209), (1MX1:A, SER1221), (1QE3:A, SER189), (2BCE:_, SER194)
CARBOXYLESTERASE_B_2	
3.CYS.SG	(1DX4:A, CYS93), (1EA5:A, CYS94), (1LLF:A, CYS97), (1MX1:A, CYS1116), (1QE3:A, CYS82), (2BCE:_, CYS80)
CHITINASE_18	
9.GLU.OE1. 9.GLU.OE2	(1EDQ:A, GLU315), (1EDT:_, GLU132), (1GOI:A, GLU144), (1ITX:A, GLU204), (1KFW:A, GLU192)
COPPER_BLUE	
11.HIS.ND1, 11.HIS.NE2	(1AAC:_, HIS95), (1BAW:A, HIS92), (1BQK:_, HIS81), (1DFE:A, HIS32), (1E30:A, HIS143), (1JER:_, HIS94), (1JZG:A, HIS117), (1KDJ:_, HIS90), (1PLC:_, HIS87), (1QHQ:A, HIS127), (2CBP:_, HIS84)
7.CYS.SG	(1AAC:_, CYS92), (1BAW:A, CYS89), (1BQK:_, CYS78), (1DFE:A, CYS27), (1E30:A, CYS138), (1JER:_, CYS89), (1JZG:A, CYS112), (1KDJ:_, CYS87), (1PLC:_, CYS84), (1QHQ:A, CYS122), (2CBP:_, CYS79)

CYTOCHROME_C	
1.CYS.SG	(19HC:A, CYS47), (19HC:A, CYS59), (19HC:A, CYS97), (19HC:A, CYS111), (19HC:A, CYS127), (19HC:A, CYS225), (19HC:A, CYS241), (19HC:A, CYS267), (19HC:A, CYS284), (1A1V:A, CYS289), (1AQE:_, CYS38), (1AQE:_, CYS86), (1AQE:_, CYS105), (1BBH:A, CYS121), (1C52:_, CYS11), (1C75:A, CYS32), (1CC5:_, CYS19), (1CNO:A, CYS14), (1COT:_, CYS15), (1CPQ:_, CYS118), (1CTJ:_, CYS15), (1DIQ:C, CYS615), (1DW0:A, CYS43), (1DXR:C, CYS87), (1DXR:C, CYS132), (1DXR:C, CYS244), (1DXR:C, CYS305), (1E29:A, CYS37), (1E2W:A, CYS21), (1E5D:A, CYS289), (1E85:A, CYS116), (1EEJ:A, CYS98), (1EEJ:A, CYS98), (1ES6:A, CYS311), (1ETP:A, CYS119), (1EXK:A, CYS14), (1EXK:A, CYS67), (1EXT:A, CYS30), (1EXT:A, CYS30), (1EZV:D, CYS101), (1FCD:C, CYS11), (1FCD:C, CYS101), (1FGJ:A, CYS79), (1FGJ:A, CYS145), (1FGJ:A, CYS172), (1FGJ:A, CYS229), (1FGJ:A, CYS239), (1FGJ:A, CYS259), (1FGJ:A, CYS310), (1FGJ:A, CYS360), (1FP0:A, CYS66), (1FS7:A, CYS168), (1FS7:A, CYS211), (1FS7:A, CYS295), (1FS7:A, CYS326), (1FT5:A, CYS11), (1FT5:A, CYS60), (1FT5:A, CYS88), (1FT5:A, CYS134), (1GKS:_, CYS14), (1GU2:A, CYS49), (1GYO:A, CYS36), (1GYO:A, CYS52), (1GYO:A, CYS80), (1H1O:A, CYS16), (1H1O:A, CYS119), (1H21:A, CYS209), (1H29:A, CYS80), (1H29:A, CYS114), (1H29:A, CYS135), (1H29:A, CYS178), (1H29:A, CYS202), (1H29:A, CYS225), (1H29:A, CYS244), (1H29:A, CYS308), (1H29:A, CYS319), (1H29:A, CYS349), (1H29:A, CYS362), (1H29:A, CYS378), (1H29:A, CYS477), (1H29:A, CYS493), (1H29:A, CYS519), (1H29:A, CYS536), (1H32:A, CYS76), (1H32:A, CYS177), (1H32:B, CYS42), (1H75:A, CYS11), (1HH5:A, CYS26), (1HH5:A, CYS49), (1HH5:A, CYS62), (1I8O:A, CYS13), (1IQC:A, CYS39), (1IQC:A, CYS183), (1JGS:A, CYS108), (1JJU:A, CYS11), (1JMX:A, CYS12), (1JNI:A, CYS58), (1JNI:A, CYS98), (1KB0:A, CYS604), (1KSS:A, CYS14), (1KSS:A, CYS36), (1KSS:A, CYS68), (1KSS:A, CYS82), (1KSS:A, CYS36), (1KV9:A, CYS591), (1M1Q:A, CYS15), (1M1Q:A, CYS35), (1M1Q:A, CYS58), (1M1Q:A, CYS75), (1MG2:D, CYS57), (1MQV:A, CYS113), (1OAH:A, CYS188), (1OAH:A, CYS230), (1OAH:A, CYS317), (1OAH:A, CYS349), (1QKS:A, CYS65), (1QL3:A, CYS14), (1Q08:A, CYS15), (1Q08:A, CYS36), (1Q08:A, CYS65), (1Q08:A, CYS79), (1YCC:_, CYS14), (2CCY:A, CYS118), (2CY3:_, CYS44), (2CY3:_, CYS92), (2CY3:_, CYS111), (3CA0:A, CYS36), (3CA0:A, CYS59), (3CA0:A, CYS96), (3CYR:_, CYS30), (3CYR:_, CYS79), (451C:_, CYS12)

4.CYS.SG	(19HC:A, CYS50), (19HC:A, CYS62), (19HC:A, CYS100), (19HC:A, CYS114), (19HC:A, CYS130), (19HC:A, CYS228), (19HC:A, CYS244), (19HC:A, CYS270), (19HC:A, CYS287), (1A1V:A, CYS292), (1AQE:_, CYS41), (1AQE:_, CYS89), (1AQE:_, CYS108), (1BBH:A, CYS124), (1C52:_, CYS14), (1C75:A, CYS35), (1CC5:_, CYS22), (1CNO:A, CYS17), (1COT:_, CYS18), (1CPQ:_, CYS121), (1CTJ:_, CYS18), (1DIQ:C, CYS618), (1DWO:A, CYS46), (1DXR:C, CYS90), (1DXR:C, CYS135), (1DXR:C, CYS247), (1DXR:C, CYS308), (1E29:A, CYS40), (1E2W:A, CYS24), (1E5D:A, CYS292), (1E85:A, CYS119), (1EEJ:A, CYS101), (1EEJ:A, CYS101), (1ES6:A, CYS314), (1ETP:A, CYS122), (1EXK:A, CYS17), (1EXK:A, CYS70), (1EXT:A, CYS33), (1EXT:A, CYS33), (1EZV:D, CYS104), (1FCD:C, CYS14), (1FCD:C, CYS104), (1FGJ:A, CYS82), (1FGJ:A, CYS148), (1FGJ:A, CYS175), (1FGJ:A, CYS232), (1FGJ:A, CYS242), (1FGJ:A, CYS262), (1FGJ:A, CYS313), (1FGJ:A, CYS363), (1FPO:A, CYS69), (1FS7:A, CYS171), (1FS7:A, CYS214), (1FS7:A, CYS298), (1FS7:A, CYS329), (1FT5:A, CYS14), (1FT5:A, CYS63), (1FT5:A, CYS91), (1FT5:A, CYS137), (1GKS:_, CYS17), (1GU2:A, CYS52), (1GYO:A, CYS39), (1GYO:A, CYS55), (1GYO:A, CYS83), (1H1O:A, CYS19), (1H1O:A, CYS122), (1H21:A, CYS212), (1H29:A, CYS83), (1H29:A, CYS117), (1H29:A, CYS138), (1H29:A, CYS181), (1H29:A, CYS205), (1H29:A, CYS228), (1H29:A, CYS247), (1H29:A, CYS311), (1H29:A, CYS322), (1H29:A, CYS352), (1H29:A, CYS365), (1H29:A, CYS381), (1H29:A, CYS480), (1H29:A, CYS496), (1H29:A, CYS522), (1H29:A, CYS539), (1H32:A, CYS79), (1H32:A, CYS180), (1H32:B, CYS45), (1H75:A, CYS14), (1HH5:A, CYS29), (1HH5:A, CYS52), (1HH5:A, CYS65), (1I8O:A, CYS16), (1IQC:A, CYS42), (1IQC:A, CYS186), (1JGS:A, CYS111), (1JJU:A, CYS14), (1JMX:A, CYS15), (1JNI:A, CYS61), (1JNI:A, CYS101), (1KBO:A, CYS607), (1KSS:A, CYS17), (1KSS:A, CYS39), (1KSS:A, CYS71), (1KSS:A, CYS85), (1KSS:A, CYS39), (1KV9:A, CYS594), (1M1Q:A, CYS18), (1M1Q:A, CYS38), (1M1Q:A, CYS61), (1M1Q:A, CYS78), (1MG2:D, CYS60), (1MQV:A, CYS116), (1OAH:A, CYS191), (1OAH:A, CYS233), (1OAH:A, CYS320), (1OAH:A, CYS352), (1QKS:A, CYS68), (1QL3:A, CYS17), (1QO8:A, CYS18), (1QO8:A, CYS39), (1QO8:A, CYS68), (1QO8:A, CYS82), (1YCC:_, CYS17), (2CCY:A, CYS121), (2CY3:_, CYS47), (2CY3:_, CYS95), (2CY3:_, CYS114), (3CAO:A, CYS39), (3CAO:A, CYS62), (3CAO:A, CYS99), (3CYR:_, CYS33), (3CYR:_, CYS82), (451C:_, CYS15)

5.HIS.ND1, 5.HIS.NE2	(19HC:A, HIS51), (19HC:A, HIS63), (19HC:A, HIS101), (19HC:A, HIS115), (19HC:A, HIS131), (19HC:A, HIS229), (19HC:A, HIS245), (19HC:A, HIS271), (19HC:A, HIS288), (1A1V:A, HIS293), (1AQE:_, HIS42), (1AQE:_, HIS90), (1AQE:_, HIS109), (1AYF:A, HIS56), (1BBH:A, HIS125), (1C52:_, HIS15), (1C75:A, HIS36), (1CC5:_, HIS23), (1CNO:A, HIS18), (1COT:_, HIS19), (1CPQ:_, HIS122), (1CTJ:_, HIS19), (1DIQ:C, HIS619), (1DWO:A, HIS47), (1DXR:C, HIS91), (1DXR:C, HIS136), (1DXR:C, HIS248), (1DXR:C, HIS309), (1E29:A, HIS41), (1E2W:A, HIS25), (1E5D:A, HIS293), (1E85:A, HIS120), (1E9M:A, HIS49), (1EEJ:A, HIS102), (1ES6:A, HIS315), (1ETP:A, HIS123), (1EXK:A, HIS18), (1EXK:A, HIS71), (1EXT:A, HIS34), (1EZV:D, HIS105), (1FCD:C, HIS15), (1FCD:C, HIS105), (1FGJ:A, HIS83), (1FGJ:A, HIS149), (1FGJ:A, HIS176), (1FGJ:A, HIS233), (1FGJ:A, HIS243), (1FGJ:A, HIS263), (1FGJ:A, HIS314), (1FGJ:A, HIS364), (1FPO:A, HIS70), (1FS7:A, HIS172), (1FS7:A, HIS215), (1FS7:A, HIS299), (1FS7:A, HIS330), (1FT5:A, HIS15), (1FT5:A, HIS64), (1FT5:A, HIS92), (1FT5:A, HIS138), (1GKS:_, HIS18), (1GU2:A, HIS53), (1GYO:A, HIS40), (1GYO:A, HIS56), (1GYO:A, HIS84), (1H1O:A, HIS20), (1H1O:A, HIS123), (1H21:A, HIS213), (1H29:A, HIS84), (1H29:A, HIS118), (1H29:A, HIS139), (1H29:A, HIS182), (1H29:A, HIS206), (1H29:A, HIS229), (1H29:A, HIS248), (1H29:A, HIS312), (1H29:A, HIS323), (1H29:A, HIS353), (1H29:A, HIS366), (1H29:A, HIS382), (1H29:A, HIS481), (1H29:A, HIS497), (1H29:A, HIS523), (1H29:A, HIS540), (1H32:A, HIS80), (1H32:A, HIS181), (1H32:B, HIS46), (1H75:A, HIS15), (1HH5:A, HIS30), (1HH5:A, HIS53), (1HH5:A, HIS66), (1I8O:A, HIS17), (1IQC:A, HIS43), (1IQC:A, HIS187), (1JGS:A, HIS112), (1JJU:A, HIS15), (1JMX:A, HIS16), (1JNI:A, HIS62), (1JNI:A, HIS102), (1KBO:A, HIS608), (1KSS:A, HIS18), (1KSS:A, HIS40), (1KSS:A, HIS72), (1KSS:A, HIS86), (1KV9:A, HIS595), (1M1Q:A, HIS19), (1M1Q:A, HIS39), (1M1Q:A, HIS62), (1M1Q:A, HIS79), (1MG2:D, HIS61), (1MQV:A, HIS117), (1OAH:A, HIS192), (1OAH:A, HIS234), (1OAH:A, HIS321), (1OAH:A, HIS353), (1QKS:A, HIS69), (1QL3:A, HIS18), (1QO8:A, HIS19), (1QO8:A, HIS40), (1QO8:A, HIS69), (1QO8:A, HIS83), (1YCC:_, HIS18), (2CCY:A, HIS122), (2CY3:_, HIS48), (2CY3:_, HIS96), (2CY3:_, HIS115), (3CAO:A, HIS40), (3CAO:A, HIS63), (3CAO:A, HIS100), (3CYR:_, HIS34), (3CYR:_, HIS83), (451C:_, HIS16)

CYTOCHROME_P450	
8.CYS.SG	(1CPT:_, CYS377), (1DZ4:A, CYS357), (1E9X:A, CYS394), (1GWI:A, CYS355), (1IO7:A, CYS317), (1JFB:A, CYS352), (1JIP:A, CYS351), (1JPZ:A, CYS400), (1LFK:A, CYS347), (1N40:A, CYS345), (1N6B:A, CYS432), (1N97:A, CYS336)
C_TYPE_LECTIN_1	
1.CYS.SG	(1BYF:A, CYS96), (1DV8:A, CYS254), (1EGI:A, CYS735), (1G1T:A, CYS90), (1H8U:A, CYS92), (1J34:A, CYS102), (1JZN:A, CYS106), (1K9J:A, CYS368), (1QDD:A, CYS115), (1TN3:_, CYS152), (2AFP:A, CYS101), (2MSB:A, CYS195)
EF_HAND	
1.ASP.OD1, 1.ASP.OD2	(1ALV:A, ASP150), (1ALV:A, ASP180), (1AUI:B, ASP30), (1AUI:B, ASP62), (1AUI:B, ASP99), (1AUI:B, ASP140), (1C07:A, ASP28), (1C7V:A, ASP98), (1C7V:A, ASP135), (1CTD:A, ASP14), (1DAV:A, ASP40), (1DGU:A, ASP108), (1DGU:A, ASP153), (1EL4:A, ASP30), (1EL4:A, ASP123), (1EL4:A, ASP159), (1EXR:A, ASP20), (1EXR:A, ASP56), (1EXR:A, ASP93), (1EXR:A, ASP129), (1FI6:A, ASP58), (1FPW:A, ASP109), (1FPW:A, ASP157), (1GAI:_, ASP403), (1HQV:A, ASP36), (1HQV:A, ASP103), (1IG5:A, ASP54), (1IJ5:A, ASP230), (1IJ5:A, ASP265), (1IJ5:A, ASP295), (1IJ5:A, ASP332), (1IRJ:A, ASP67), (1JBA:A, ASP69), (1JBA:A, ASP105), (1JBA:A, ASP158), (1JFJ:A, ASP10), (1JFJ:A, ASP46), (1JFJ:A, ASP85), (1JFJ:A, ASP117), (1JJ2:C, ASP9), (1K3I:A, ASP75), (1K8U:A, ASP61), (1K94:A, ASP132), (1K9U:A, ASP13), (1K9U:A, ASP48), (1M1X:A, ASP60), (1MR8:A, ASP59), (1OQP:A, ASP111), (1OQP:A, ASP147), (1PG4:A, ASP101), (1PSR:A, ASP62), (1QLS:A, ASP66), (1REC:_, ASP110), (1SRA:_, ASP257), (1WDC:B, ASP28), (2PVB:A, ASP51), (2PVB:A, ASP90), (2SAS:_, ASP19), (2SAS:_, ASP70), (2SAS:_, ASP115), (2SCP:A, ASP16), (2SCP:A, ASP104), (2SCP:A, ASP138)
3.ASN.ND2, 3.ASN.OD1	(1C7V:A, ASN100), (1CTD:A, ASN16), (1DAV:A, ASN42), (1EL4:A, ASN32), (1FPW:A, ASN111), (1FPW:A, ASN159), (1IG5:A, ASN56), (1IJ5:A, ASN232), (1IRJ:A, ASN69), (1JBA:A, ASN71), (1JBA:A, ASN160), (1JFJ:A, ASN12), (1JFJ:A, ASN119), (1JJ2:C, ASN11), (1K3I:A, ASN77), (1K8U:A, ASN63), (1K9U:A, ASN15), (1MR8:A, ASN61), (1OQP:A, ASN149), (1PSR:A, ASN64), (2SAS:_, ASN21), (2SAS:_, ASN72), (2SCP:A, ASN106), (2SCP:A, ASN140)

3.ASP.OD1, 3.ASP.OD2	(1ALV:A, ASP152), (1ALV:A, ASP182), (1AUI:B, ASP32), (1AUI:B, ASP64), (1AUI:B, ASP101), (1AUI:B, ASP142), (1C07:A, ASP30), (1C7V:A, ASP137), (1DGU:A, ASP110), (1DGU:A, ASP155), (1EL4:A, ASP125), (1EL4:A, ASP161), (1EXR:A, ASP22), (1EXR:A, ASP58), (1EXR:A, ASP95), (1EXR:A, ASP131), (1FI6:A, ASP60), (1HQV:A, ASP38), (1HQV:A, ASP105), (1IJ5:A, ASP267), (1IJ5:A, ASP334), (1JBA:A, ASP107), (1JFJ:A, ASP48), (1JFJ:A, ASP87), (1K94:A, ASP134), (1K9U:A, ASP50), (1OQP:A, ASP113), (1QLS:A, ASP68), (1REC:_, ASP112), (1SRA:_, ASP259), (1WDC:B, ASP30), (2PVB:A, ASP53), (2PVB:A, ASP92), (2SCP:A, ASP18)
5.ASN.ND2, 5.ASN.OD1	(1AUI:B, ASN66), (1C7V:A, ASN139), (1EL4:A, ASN34), (1EXR:A, ASN60), (1EXR:A, ASN97), (1IJ5:A, ASN234), (1JBA:A, ASN109), (1JFJ:A, ASN50), (1K3I:A, ASN79), (1REC:_, ASN114)
5.ASP.OD1, 5.ASP.OD2	(1AUI:B, ASP103), (1AUI:B, ASP144), (1C07:A, ASP32), (1C7V:A, ASP102), (1CTD:A, ASP18), (1DAV:A, ASP44), (1DGU:A, ASP112), (1DGU:A, ASP157), (1EXR:A, ASP24), (1EXR:A, ASP133), (1FI6:A, ASP62), (1FPW:A, ASP113), (1FPW:A, ASP161), (1IG5:A, ASP58), (1IRJ:A, ASP71), (1JBA:A, ASP73), (1JBA:A, ASP162), (1JFJ:A, ASP14), (1JFJ:A, ASP89), (1JFJ:A, ASP121), (1JJ2:C, ASP13), (1K8U:A, ASP65), (1K9U:A, ASP17), (1K9U:A, ASP52), (1MR8:A, ASP63), (1OQP:A, ASP151), (1PSR:A, ASP66), (1QLS:A, ASP70), (1SRA:_, ASP261), (1WDC:B, ASP32), (2PVB:A, ASP94), (2SAS:_, ASP23), (2SAS:_, ASP74), (2SAS:_, ASP119), (2SCP:A, ASP20), (2SCP:A, ASP108), (2SCP:A, ASP142)
5.SER.OG	(1ALV:A, SER184), (1AUI:B, SER34), (1EL4:A, SER127), (1EL4:A, SER163), (1HQV:A, SER40), (1HQV:A, SER107), (1IJ5:A, SER299), (1IJ5:A, SER336), (1K94:A, SER136), (1OQP:A, SER115), (1PG4:A, SER105), (2PVB:A, SER55)
7.GLU.OE1, 7.GLU.OE2	(1AUI:B, GLU68), (1GAI:_, GLU409), (1IG5:A, GLU60), (1JFJ:A, GLU52), (1JJ2:C, GLU15), (1K8U:A, GLU67), (1OQP:A, GLU153)
7.LYS.NZ	(1ALV:A, LYS156), (1EL4:A, LYS36), (1JFJ:A, LYS91), (1K9U:A, LYS19), (1PSR:A, LYS68)
7.THR.OG1	(1ALV:A, THR186), (1DGU:A, THR114), (1DGU:A, THR159), (1EL4:A, THR129), (1EXR:A, THR26), (1EXR:A, THR62), (1IJ5:A, THR236), (1JBA:A, THR75), (1K94:A, THR138), (1OQP:A, THR117), (1REC:_, THR116)
7.TYR.OH	(1AUI:B, TYR105), (1CTD:A, TYR20), (1FPW:A, TYR115), (1FPW:A, TYR163), (1JFJ:A, TYR123), (1SRA:_, TYR263)
9.ASN.ND2, 9.ASN.OD1	(1DAV:A, ASN48), (1DGU:A, ASN116), (1DGU:A, ASN161), (1EXR:A, ASN137), (1K8U:A, ASN69), (1MR8:A, ASN67)

9.ASP.OD1, 9.ASP.OD2	(1AUI:B, ASP70), (1C7V:A, ASP106), (1C7V:A, ASP143), (1CTD:A, ASP22), (1EL4:A, ASP167), (1EXR:A, ASP64), (1HQV:A, ASP111), (1JBA:A, ASP77), (1JBA:A, ASP113), (1JFJ:A, ASP54), (1JJ2:C, ASP17), (1K9U:A, ASP56), (1OQP:A, ASP155), (1PSR:A, ASP70), (1QLS:A, ASP74), (2SAS:_, ASP123)
9.SER.OG	(1AUI:B, SER38), (1AUI:B, SER107), (1AUI:B, SER148), (1C07:A, SER36), (1EXR:A, SER101), (1GAI:_, SER411), (1HQV:A, SER44), (1IG5:A, SER62), (1IJ5:A, SER238), (1IJ5:A, SER303), (1IJ5:A, SER340), (1IRJ:A, SER75), (1JBA:A, SER166), (1JFJ:A, SER18), (1K9U:A, SER21), (1PG4:A, SER109), (1REC:_, SER118), (1WDC:B, SER36), (2SAS:_, SER78), (2SCP:A, SER112), (2SCP:A, SER146)
9.THR.OG1	(1EL4:A, THR38), (1EL4:A, THR131), (1EXR:A, THR28), (1FI6:A, THR66), (1FPW:A, THR117), (1FPW:A, THR165), (1JFJ:A, THR93), (1JFJ:A, THR125), (1OQP:A, THR119), (2SCP:A, THR24)
12.TYR.OH	(1IJ5:A, TYR277), (1JBA:A, TYR81), (1K8U:A, TYR73), (2SAS:_, TYR82), (2SCP:A, TYR116)
	EGF_1
1.CYS.SG	(1AUT:L, CYS78), (1AUT:L, CYS78), (1B6E:_, CYS59), (1EDM:B, CYS71), (1EDM:B, CYS71), (1EDM:B, CYS71), (1G1T:A, CYS142), (1G1T:A, CYS142), (1HAE:_, CYS34), (1HAE:_, CYS34), (1IOX:A, CYS32), (1IOX:A, CYS32), (1JL9:A, CYS31), (1JL9:A, CYS31), (1KLO:_, CYS143), (1KLO:_, CYS143), (1LK9:A, CYS39), (1M1X:B, CYS547), (1M1X:B, CYS547), (1O9A:A, CYS45), (1O9A:A, CYS45), (1TPG:_, CYS73), (1TPG:_, CYS73), (1XDT:R, CYS132), (1XDT:R, CYS132)
3.CYS.SG	(1AUT:L, CYS80), (1AUT:L, CYS80), (1B6E:_, CYS61), (1EDM:B, CYS73), (1EDM:B, CYS73), (1G1T:A, CYS144), (1G1T:A, CYS144), (1HAE:_, CYS36), (1HAE:_, CYS36), (1IOX:A, CYS34), (1IOX:A, CYS34), (1JL9:A, CYS33), (1JL9:A, CYS33), (1KLO:_, CYS145), (1KLO:_, CYS145), (1LK9:A, CYS41), (1M1X:B, CYS549), (1M1X:B, CYS549), (1O9A:A, CYS47), (1O9A:A, CYS47), (1TPG:_, CYS75), (1TPG:_, CYS75), (1XDT:R, CYS134), (1XDT:R, CYS134)
7.CYS.SG	(1AUT:L, CYS89), (1AUT:L, CYS89), (1B6E:_, CYS70), (1EDM:B, CYS82), (1EDM:B, CYS82), (1G1T:A, CYS153), (1G1T:A, CYS153), (1HAE:_, CYS45), (1HAE:_, CYS45), (1IOX:A, CYS43), (1IOX:A, CYS43), (1JL9:A, CYS42), (1JL9:A, CYS42), (1KLO:_, CYS154), (1KLO:_, CYS154), (1LK9:A, CYS50), (1M1X:B, CYS558), (1O9A:A, CYS56), (1O9A:A, CYS56), (1TPG:_, CYS84), (1TPG:_, CYS84), (1XDT:R, CYS143), (1XDT:R, CYS143)

EGF_2	
1.CYS.SG	(1COU:A, CYS50), (1DX5:I, CYS407), (1EMO:_, CYS2151), (1EMO:_, CYS2151), (1EXT:A, CYS137), (1FJS:L, CYS109), (1HZ8:A, CYS25), (1HZ8:A, CYS64), (1KLI:L, CYS112), (1NTO:A, CYS146)
3.CYS.SG	(1COU:A, CYS52), (1DX5:I, CYS409), (1EMO:_, CYS2153), (1EXT:A, CYS139), (1FJS:L, CYS111), (1HZ8:A, CYS27), (1HZ8:A, CYS66), (1KLI:L, CYS114), (1NTO:A, CYS148)
8.CYS.SG	(1COU:A, CYS64), (1DX5:I, CYS421), (1EMO:_, CYS2164), (1EXT:A, CYS150), (1FJS:L, CYS124), (1HZ8:A, CYS39), (1HZ8:A, CYS79), (1KLI:L, CYS127), (1NTO:A, CYS161)
GLYCOSYL_HYDROL_F5	
7.GLU.OE1, 7.GLU.OE2	(1BQC:A, GLU128), (1CZ1:A, GLU192), (1ECE:A, GLU162), (1EDG:_, GLU170), (1EGZ:A, GLU133), (7A3H:A, GLU139)
GLYCOSYL_HYDROL_F10	
7.GLU.OE1, 7.GLU.OE2	(1CLX:A, GLU246), (1HIZ:A, GLU266), (1I1W:A, GLU237), (1L3I:A, GLU158), (1L5A:A, GLU48), (1XYZ:A, GLU754)
HIPIP	
1.CYS.SG	(1HLQ:A, CYS54), (1HPI:_, CYS51), (1ISU:A, CYS40), (1IUA:A, CYS61), (2HIP:A, CYS48)
7.CYS.SG	(1HLQ:A, CYS68), (1HPI:_, CYS65), (1ISU:A, CYS55), (1IUA:A, CYS75), (2HIP:A, CYS64)
HMA_1	
5.CYS.SG	(1AFJ:_, CYS14), (1AWO:_, CYS14), (1CC8:A, CYS15), (1CPZ:A, CYS11), (1FEO:A, CYS12), (1FVQ:A, CYS13), (1JWW:A, CYS14), (1KOV:A, CYS13), (1MWZ:A, CYS15)
8.CYS.SG	(1AFJ:_, CYS17), (1AWO:_, CYS17), (1CC8:A, CYS18), (1CPZ:A, CYS14), (1FEO:A, CYS15), (1FVQ:A, CYS16), (1JWW:A, CYS17), (1KOV:A, CYS16), (1MWZ:A, CYS18)
IG_MHC	
3.CYS.SG	(1F0X:A, CYS492), (1FNG:A, CYS163), (1FNG:B, CYS173), (1FP5:A, CYS418), (1FP5:A, CYS524), (1GPP:A, CYS75), (1HDM:A, CYS173), (1HDM:B, CYS167), (1K5N:A, CYS259), (1K5N:B, CYS80), (1L6X:A, CYS425), (1MV8:A, CYS213), (1NCW:L, CYS194), (3FRU:A, CYS254), (8FAB:A, CYS193)

IMP_1	
4.ASP.OD1, 4.ASP.OD2	(1G0H:A, ASP81), (1INP:_, ASP153), (1KA1:A, ASP142), (1LBV:A, ASP82), (2HHM:A, ASP90)
KAZAL	
1.CYS.SG	(1LDT:L, CYS6I), (1PCE:_, CYS20), (1SGP:I, CYS16), (1TBR:R, CYS8), (1TGS:I, CYS16)
3.CYS.SG	(1LDT:L, CYS14I), (1PCE:_, CYS28), (1SGP:I, CYS24), (1TBR:R, CYS16), (1TGS:I, CYS24)
7.CYS.SG	(1LDT:L, CYS25I), (1PCE:_, CYS39), (1SGP:I, CYS35), (1TBR:R, CYS27), (1TGS:I, CYS35)
9.CYS.SG	(1LDT:L, CYS29I), (1PCE:_, CYS42), (1SGP:I, CYS38), (1TBR:R, CYS31), (1TGS:I, CYS38)
LIPASE_SER	
7.SER.OG	(1BU8:A, SER152), (1CVL:_, SER87), (1G66:A, SER90), (1HLG:A, SER153), (1JFR:A, SER131), (1MNA:A, SER148), (1TIB:_, SER146), (3TGL:_, SER144)
LIPOYL	
9.LYS.NZ	(1FYC:_, LYS50), (1GHK:_, LYS42), (1GW5:B, LYS78), (1HTP:_, LYS63), (1K8M:A, LYS45), (1LAC:_, LYS42), (1QJO:A, LYS41)
L_LDH	
4.HIS.ND1, 4.HIS.NE2	(1EZ4:A, HIS193), (1HYE:A, HIS178), (1HYH:A, HIS198), (1LDM:_, HIS193), (1LLD:A, HIS180)
PA2_HIS	
4.HIS.ND1, 4.HIS.NE2	(1EN2:A, HIS67), (1G4I:A, HIS48), (1LE6:A, HIS46), (1MC2:A, HIS1048), (1POC:_, HIS34)
PEROXIDASE_1	
8.HIS.ND1, 8.HIS.NE2	(1ARU:_, HIS184), (1GPE:A, HIS162), (1GWU:A, HIS170), (1JDR:A, HIS175), (1MWV:A, HIS279), (1N62:C, HIS35), (1OAF:A, HIS163)
PEROXIDASE_2	
8.HIS.ND1, 8.HIS.NE2	(1ARU:_, HIS56), (1GWU:A, HIS42), (1JDR:A, HIS52), (1MWV:A, HIS112), (1OAF:A, HIS42)

PHOSPHOPANTETHEINE	
6.SER.OG	(1AF8:_, SER42), (1DNY:A, SER45), (1G1K:A, SER68), (1KB0:A, SER65), (1L0I:A, SER36), (1N8L:A, SER38), (1NW1:A, SER225), (1QQE:A, SER93), (2UAG:A, SER116)
PROTEIN_KINASE_ST	
5.ASP.OD1, 5.ASP.OD2	(1A06:_, ASP141), (1APM:E, ASP166), (1B6C:B, ASP333), (1CSN:_, ASP131), (1F3M:C, ASP389), (1GNG:A, ASP181), (1GZ8:A, ASP127), (1H1W:A, ASP205), (1HOW:A, ASP294), (1JKS:A, ASP139), (1KOB:A, ASP174), (1KWP:A, ASP186), (1M2R:A, ASP156), (1MU0:A, ASP256), (1NVR:A, ASP130), (1O6L:A, ASP275), (1O6Y:A, ASP138), (1OMW:A, ASP317), (1PHK:_, ASP149), (1PME:_, ASP149)
PROTEIN_KINASE_TYR	
5.ASP.OD1, 5.ASP.OD2	(1FGK:A, ASP623), (1M14:A, ASP813), (1P4O:A, ASP1105), (1QPC:A, ASP364), (1TKI:A, ASP144)
PTS_HPR_SER	
5.SER.OG	(1D8C:A, SER333), (1G61:A, SER2177), (1IU0:A, SER56), (1KSS:A, SER446), (1PCH:_, SER46), (1PTF:_, SER46)
RNASE_T2_1	
4.HIS.ND1, 4.HIS.NE2	(1BOL:A, HIS46), (1DIX:A, HIS39), (1IOO:A, HIS32), (1IQQ:A, HIS33), (1UCA:A, HIS34)
SHIGA_RICIN	
5.GLU.OE1, 5.GLU.OE2	(1D6A:A, GLU176), (1DM0:A, GLU167), (1GGP:A, GLU164), (1HWM:A, GLU163), (1IFT:_, GLU177), (1MRJ:_, GLU160), (1QI7:A, GLU176)
8.ARG.NE, 8.ARG.NH1, 8.ARG.NH2	(1D6A:A, ARG179), (1DM0:A, ARG170), (1GGP:A, ARG167), (1HWM:A, ARG166), (1IFT:_, ARG180), (1MRJ:_, ARG163), (1QI7:A, ARG179)
SMALL_CYTOKINES_CC	
1.CYS.SG	(1B3A:A, CYS10), (1CM9:A, CYS14), (1DOK:A, CYS11), (1EL0:A, CYS11), (1G2T:A, CYS10), (1M8A:A, CYS6)
2.CYS.SG	(1B3A:A, CYS11), (1CM9:A, CYS15), (1DOK:A, CYS12), (1EL0:A, CYS12), (1G2T:A, CYS11), (1M8A:A, CYS7)
11.CYS.SG	(1B3A:A, CYS34), (1CM9:A, CYS38), (1DOK:A, CYS36), (1EL0:A, CYS35), (1G2T:A, CYS34), (1M8A:A, CYS32)
12.CYS.SG	(1B3A:A, CYS50), (1CM9:A, CYS54), (1DOK:A, CYS52), (1EL0:A, CYS51), (1G2T:A, CYS50), (1M8A:A, CYS48)

SNAKE_TOXIN	
2.CYS.SG	(1FAS:_, CYS39), (1FF4:A, CYS42), (1JGK:A, CYS43), (1TGX:A, CYS38), (2CTX:_, CYS41), (3EBX:_, CYS41)
4.CYS.SG	(1FAS:_, CYS41), (1FF4:A, CYS46), (1JGK:A, CYS47), (1TGX:A, CYS42), (2CTX:_, CYS45), (3EBX:_, CYS43)
7.CYS.SG	(1FAS:_, CYS52), (1FF4:A, CYS57), (1JGK:A, CYS59), (1TGX:A, CYS53), (2CTX:_, CYS56), (3EBX:_, CYS54)
8.CYS.SG	(1FAS:_, CYS53), (1FF4:A, CYS58), (1JGK:A, CYS60), (1TGX:A, CYS54), (2CTX:_, CYS57), (3EBX:_, CYS55)
SUBTILASE_ASP	
5.ASP.OD1, 5.ASP.OD2	(1CJY:A, ASP549), (1GCI:_, ASP32), (1IC6:A, ASP39), (1MG7:A, ASP58), (1OT5:A, ASP175), (2RSL:A, ASP94)
THIOL_PROTEASE_ASN	
6.ASN.ND2, 6.ASN.OD1	(1CS8:A, ASN187), (1DEU:A, ASN200), (1GMY:A, ASN219), (1IWD:A, ASN178), (1ME4:A, ASN175)
THIOL_PROTEASE_HIS	
3.HIS.ND1, 3.HIS.NE2	(1CS8:A, HIS163), (1CV8:_, HIS120), (1EZI:A, HIS88), (1GMY:A, HIS199), (1ME4:A, HIS159), (1NST:A, HIS731), (1QNT:A, HIS29), (3GCB:_, HIS369)
THIOREDOXIN	
8.CYS.SG	(1ERV:_, CYS32), (1F9M:A, CYS46), (1FVK:A, CYS30), (1JFU:A, CYS72), (1KNG:A, CYS92), (1MEK:_, CYS36), (2TRX:A, CYS32)
11.CYS.SG	(1ERV:_, CYS35), (1F9M:A, CYS49), (1FVK:A, CYS33), (1JFU:A, CYS75), (1KNG:A, CYS95), (1MEK:_, CYS39), (2TRX:A, CYS35)
TRYPSIN_HIS	
5.HIS.ND1, 5.HIS.NE2	(1AOJ:A, HIS57), (1BIO:_, HIS57), (1BQY:A, HIS57), (1C5M:D, HIS57), (1DLE:A, HIS57), (1EAX:A, HIS57), (1EQ9:A, HIS57), (1GDN:A, HIS57), (1GVK:B, HIS57), (1GVZ:A, HIS57), (1KLI:H, HIS57), (1LTO:A, HIS57), (1M9U:A, HIS57), (1NN6:A, HIS60), (1QQ4:A, HIS36), (1SGP:E, HIS57), (2HLC:A, HIS57)

TRYPSIN_SER	
6.SER.OG	(1A0J:A, SER195), (1BIO:_, SER195), (1C5M:D, SER195), (1DLE:A, SER195), (1EAX:A, SER195), (1ELV:A, SER617), (1EQ9:A, SER195), (1GDN:A, SER195), (1GVK:B, SER195), (1GVZ:A, SER195), (1KLI:H, SER195), (1LTO:A, SER195), (1M9U:A, SER195), (1NN6:A, SER197), (1QQ4:A, SER143), (1SGP:E, SER195), (2HLC:A, SER195)
TYR_PHOSPHATASE_1	
3.CYS.SG	(1D5R:A, CYS124), (1G4U:S, CYS481), (1I9S:A, CYS126), (1JLN:A, CYS480), (1LAR:A, CYS1522), (1LAR:A, CYS1813), (1VHR:A, CYS124), (2SHP:A, CYS459)
UBIQUITIN_CONJUGAT_1	
10.CYS.SG	(1C4Z:D, CYS86), (1JAT:A, CYS87), (1PZV:A, CYS88), (1U9A:A, CYS93), (2AAK:_, CYS88), (2E2C:_, CYS114)
ZINC_FINGER_C2H2_1	
1.CYS.SG	(1A1I:A, CYS107), (1BHI:_, CYS9), (1E53:A, CYS360), (1FN9:A, CYS51), (1NCS:_, CYS34), (1NJQ:A, CYS8), (1PAA:_, CYS134), (1TF3:A, CYS15), (1TF3:A, CYS75), (1UBD:C, CYS298), (1UBD:C, CYS327), (1YUJ:A, CYS36), (2DRP:A, CYS113), (2DRP:A, CYS143), (2GLI:A, CYS106), (2GLI:A, CYS202)
3.CYS.SG	(1A1I:A, CYS112), (1BHI:_, CYS14), (1E53:A, CYS363), (1FN9:A, CYS54), (1NCS:_, CYS39), (1NJQ:A, CYS11), (1PAA:_, CYS137), (1TF3:A, CYS20), (1TF3:A, CYS80), (1UBD:C, CYS303), (1UBD:C, CYS330), (1YUJ:A, CYS39), (2DRP:A, CYS116), (2DRP:A, CYS146), (2GLI:A, CYS111), (2GLI:A, CYS207)
7.HIS.ND1, 7.HIS.NE2	(1A1I:A, HIS125), (1BHI:_, HIS27), (1E53:A, HIS376), (1FN9:A, HIS67), (1KSS:A, HIS52), (1NCS:_, HIS52), (1NJQ:A, HIS24), (1PAA:_, HIS150), (1TF3:A, HIS33), (1TF3:A, HIS93), (1UBD:C, HIS316), (1UBD:C, HIS343), (1YUJ:A, HIS52), (2DRP:A, HIS129), (2DRP:A, HIS159), (2GLI:A, HIS124), (2GLI:A, HIS220)
9.HIS.ND1, 9.HIS.NE2	(1A1I:A, HIS129), (1BHI:_, HIS31), (1E53:A, HIS380), (1FN9:A, HIS71), (1KSS:A, HIS58), (1NCS:_, HIS56), (1NJQ:A, HIS28), (1PAA:_, HIS155), (1TF3:A, HIS37), (1TF3:A, HIS98), (1UBD:C, HIS320), (1UBD:C, HIS347), (1YUJ:A, HIS57), (2DRP:A, HIS134), (2DRP:A, HIS164), (2GLI:A, HIS129), (2GLI:A, HIS225)

ZINC_PROTEASE	
3.HIS.ND1, 3.HIS.NE2	(1AST:_, HIS92), (1BKC:A, HIS405), (1DI1:A, HIS191), (1DMT:A, HIS583), (1EB6:A, HIS128), (1EPW:A, HIS229), (1EZM:_, HIS140), (1FX7:A, HIS219), (1HS6:A, HIS295), (1I1I:P, HIS474), (1J7N:A, HIS686), (1K9X:A, HIS269), (1KAP:P, HIS176), (1KEI:A, HIS142), (1KUF:A, HIS144), (1LML:_, HIS264), (1O8A:A, HIS383), (3FAP:B, HIS113)
4.GLU.OE1, 4.GLU.OE2	(1AST:_, GLU93), (1BKC:A, GLU406), (1DI1:A, GLU192), (1DMT:A, GLU584), (1EB6:A, GLU129), (1EPW:A, GLU230), (1EZM:_, GLU141), (1FX7:A, GLU220), (1HS6:A, GLU296), (1I1I:P, GLU475), (1J7N:A, GLU687), (1K9X:A, GLU270), (1KAP:P, GLU177), (1KEI:A, GLU143), (1KUF:A, GLU145), (1LML:_, GLU265), (1O8A:A, GLU384), (3FAP:B, GLU114)
7.HIS.ND1, 7.HIS.NE2	(1AST:_, HIS96), (1BKC:A, HIS409), (1DI1:A, HIS195), (1DMT:A, HIS587), (1EB6:A, HIS132), (1EPW:A, HIS233), (1EZM:_, HIS144), (1FX7:A, HIS223), (1HS6:A, HIS299), (1I1I:P, HIS478), (1J7N:A, HIS690), (1K9X:A, HIS273), (1KAP:P, HIS180), (1KEI:A, HIS146), (1KUF:A, HIS148), (1LML:_, HIS268), (1O8A:A, HIS387), (3FAP:B, HIS117)

A.3 Test sets for method comparison

We generated test sets based on PROSITE for comparing SeqFEATURE to other methods as described in Section 3.1.3. The methods we compared were PROSITE, Gene3D, Pfam, and HMMPanther (sequence-based); and SSM and 3D templates (structure-based). Below are the full test sets for PROSITE, and the subsets that were used to compare against the other methods.

Table A.3: PROSITE-derived true positive test set.

2FE2S_FERREDOXIN	(O33818, 1RM6:C), (P00216, 1E0Z:A), (P00221, 1A70:A), (P00235, 1FRR:A), (P00237, 1WRI:A), (P00246, 4FXC:A), (P00248, 1RFK:A), (P00250, 1FXI:A), (P0A3C7, 1FXA:A), (P0A3C9, 1ROE:A), (P11053, 1FRD:A), (P21912, 1ZOY:B), (P22985, 1WYG:A), (P27320, 1DOX:A), (P27787, 1GAQ:B), (P56408, 1AWD:A), (Q46509, 1SIJ:A)

4FE4S_FERREDOXIN	(P21912, 1ZOY:B), (P07485, 1DWL:A), (Q45560, 1BC6:_), (P00208, 1BLU:_), (P00195, 1CLF:_), (P00209, 1F2G:_), (P00193, 1DUR:A), (P31087, 1K0T:A)
AA_TRANSFER_CLASS_1	(P18485, 1IAX:A), (P00504, 2CST:A), (P23542, 1YAA:A), (P00508, 1AMA:A), (Q56232, 1B5O:A), (P04693, 3TAT:A), (P33447, 1BW0:A)
AA_TRANSFER_CLASS_3	(P22256, 1SF2:A), (Q7RT90, 1Z7D:A), (Q8DLK8, 2CFB:A), (Q93R93, 1VEF:A)
ADH_SHORT	(O70351, 1E3W:A), (O75828, 2HRB:A), (P00334, 1MG5:A), (P09417, 1HDR:A), (P0A5Y5, 1UZN:A), (P0AEK2, 1Q7B:A), (P11348, 1DHR:A), (P15047, 2FWM:X), (P15428, 2GDZ:A), (P16152, 1WMA:A), (P16544, 1W4Z:A), (P22414, 2ET6:A), (P28845, 1XU7:A), (P42556, 1P33:A), (P50162, 1AE1:A), (P50172, 1Y5M:A), (P51659, 1ZBQ:A), (P69167, 1NFF:A), (Q12634, 1DOH:A), (Q56840, 2CFC:A), (Q59787, 1K2W:A), (Q6QLL4, 1XSE:A), (Q6UWP2, 1XG5:A), (Q7Z4W1, 1WNT:A), (Q949M3, 2CDH:G), (Q99714, 1U7T:A), (Q9ASX2, 1XQ1:A), (Q9BPX1, 1YDE:A), (Q9BUT1, 2AG5:A)
ADH_ZINC	(O57380, 1P0C:A), (O58389, 2D8A:A), (P00325, 1DEH:A), (P00326, 1HT0:A), (P00328, 1EE2:A), (P00330, 2HCY:A), (P07327, 1HSO:A), (P11766, 1M6H:A), (P14941, 1BXZ:A), (P26325, 1CDO:A), (P35630, 1Y9A:A), (P40394, 1AGN:A), (P42328, 1RJW:A), (P75691, 1UUF:A), (Q00796, 1PL6:A), (Q04894, 1PIW:A), (Q9QYY9, 1E3E:A)
ADX	(P00259, 1OQQ:A)
ASP_PROTEASE	(P00790, 1FLH:A), (P00791, 1F34:A), (P00794, 1CMS:A), (P00796, 1SMR:A), (P00798, 1APT:E), (P00799, 2ASI:A), (P03355, 1RW3:A), (P03366, 1A9M:A), (P03367, 1A8G:A), (P04024, 1NSO:A), (P04584, 1HII:A), (P04585, 1A30:A), (P04587, 1BDQ:A), (P05896, 1AZ5:A), (P07267, 1DP5:A), (P07339, 1LYA:A), (P07570, 2D4M:A), (P12497, 4PHV:A), (P12499, 1HXW:A), (P14091, 1TZS:A), (P17576, 1WKR:A), (P20142, 1AVF:A), (P28871, 1EAG:A), (P35963, 1K6C:A), (P42210, 1QDM:A), (P56272, 1AM5:A), (Q12567, 1IBQ:A)
ASX_HYDROXYL	(P00736, 1APQ:A), (P00743, 1WHE:A), (P00742, 1IOE:L), (P08709, 1DVA:L), (O00187, 1SZB:A), (P46531, 1TOZ:A), (P07225, 1Z6C:A), (Q9UHX3, 2BO2:A), (Q9JJS8, 1NT0:A), (P10493, 1GL4:A)
BETA_LACTAMASE_A	(P0AD64, 1ONG:A), (P30896, 1N9B:A), (P52664, 1HZO:A), (P00808, 1I2S:A), (Q59517, 2CC1:A), (P16897, 1G68:A), (Q47066, 1BZA:A), (P0C5C1, 2GDN:A)
BETA_LACTAMASE_B_1	(P26918, 1X8G:A)

BPTI_KUNITZ_1	(P25660, 1JC6:A), (P00979, 1DEM:A), (P00981, 1DTK:A), (O43278, 1YC0:I), (P10646, 1IRH:A), (P48307, 1ZR0:B), (P81658, 1BF0:A)
CARBOXYLESTERASE_B_1	(P22303, 1B41:A), (P21836, 1C2B:A), (P19835, 1F6W:A), (P06276, 1P0I:A), (P12337, 1K4Y:A), (P20261, 1CRL:A), (P32946, 1GZ7:A), (P22394, 1THG:A)
CARBOXYLESTERASE_B_2	(P22303, 1B41:A), (P21836, 1C2B:A), (P19835, 1F6W:A), (P06276, 1P0I:A), (P12337, 1K4Y:A), (P20261, 1CRL:A), (P32946, 1GZ7:A), (P22394, 1THG:A)
CHITINASE_18	(P54196, 1D2K:A), (P07254, 1CTN:A), (P11797, 1O6I:A), (Q13231, 1HKJ:A), (P23472, 1KQY:A)
COPPER_BLUE	(P22365, 1ID2:A), (P04377, 1PAZ:A), (P04171, 1PMY:A), (P80401, 1ADW:A), (P56547, 1RKR:A), (P56275, 1DYZ:A), (P12335, 1CU0:A), (P00280, 1A4A:A), (P80546, 1J0I:A), (P34097, 1NW0:A), (P80728, 1WS7:A), (P46444, 1TU2:A), (Q3M9H8, 1FA4:A), (P18068, 2PLT:A), (P07465, 7PCY:A), (P17341, 1PLA:A), (P00287, 9PCY:A), (P50057, 1B3I:A), (P07030, 1BY0:A), (P00289, 1AG6:A), (P55020, 1BXU:A), (P21697, 1JXD:A), (P56274, 1IUZ:A), (P42849, 1X9R:A), (Q8RMH6, 2AAN:A), (P00281, 2IAA:A), (P0C178, 2GIM:A)
CYTOCHROME_P450	(P00178, 1P05:A), (P08684, 1TQN:A), (P10632, 1PQ2:A), (P10635, 2F9Q:A), (P11509, 1Z10:A), (P11712, 1OG2:A), (Q8RN03, 1UED:A)
C_TYPE_LECTIN	(O93427, 1UMR:C), (P06734, 1T8C:A), (P07897, 1TDQ:B), (P08427, 1R13:A), (P08661, 1BV4:A), (P11226, 1HUP:A), (P16109, 1G1Q:A), (P22029, 1FVU:A), (P23807, 1BJ3:A), (P35247, 1B08:A), (Q06141, 1UV0:A)

EF_HAND	(O14815, 1ZIV:A), (O16305, 1OOJ:A), (O43745, 2BEC:A), (O95843, 2GGZ:A), (P02586, 1A2X:A), (P02592, 1EJ3:A), (P02609, 1M8Q:B), (P02618, 1B8C:A), (P02621, 1A75:A), (P02625, 1G33:A), (P02628, 1PVA:A), (P02631, 1OMD:A), (P02632, 1CB1:A), (P02638, 1MHO:A), (P04271, 1UWO:A), (P04353, 1QS7:A), (P04631, 1DT7:A), (P04632, 1KFU:S), (P05044, 1GJY:A), (P05094, 1SJJ:A), (P05938, 2HQ8:A), (P06495, 1NYA:A), (P06704, 2DOQ:A), (P06787, 1F55:A), (P07164, 1SL8:A), (P07171, 2F33:A), (P07384, 1ZCM:A), (P08053, 2BL0:B), (P09860, 1AJ4:A), (P10688, 1DJG:A), (P17655, 1KFU:L), (P17928, 1VRK:A), (P20472, 1RJV:A), (P24480, 1NSH:A), (P26447, 1M31:A), (P27482, 1GGZ:A), (P28583, 1S6I:A), (P30563, 5PAL:A), (P30626, 1JUO:A), (P30801, 1A03:A), (P32930, 1TTX:A), (P35467, 1K2H:A), (P41208, 1M39:A), (P56503, 1BU3:A), (P61023, 2CT9:A), (P61602, 1BJF:A), (P62144, 2BE6:B), (P62149, 1AHR:A), (P62152, 1MXE:A), (P62155, 1CFF:A), (P62157, 1A29:A), (P62161, 1G4Y:A), (P62166, 1G8I:B), (P62204, 1QIW:A), (P63099, 1TCO:B), (P63316, 1AP4:A), (P63317, 1J1D:A), (Q02818, 1SNL:A), (Q06850, 2AAO:A), (Q07009, 1DF0:A), (Q09196, 1GGW:A), (Q25088, 1YX7:A), (Q39419, 1H4B:A), (Q5THR3, 1WLZ:A), (Q64537, 1AJ5:A), (Q84V36, 1PMZ:A), (Q8NFH8, 1IQ3:A), (Q8R426, 1S6C:A), (Q9NZI2, 1S1E:A), (Q9NZT1, 2B1U:A), (Q9SDJ0, 1QTX:A), (Q9SRP5, 1TIZ:A), (P97571, 1KXR:A), (P02588, 1AVS:A)
EGF_1	(Q01594, 1LK9:_), (P01132, 1A3P:A), (P00743, 1WHE:A), (P00742, 1XKB:A), (P08709, 1FAK:A), (P00741, 1PFX:L), (P16293, 1X7A:C), (P16109, 1FSB:A), (P46531, 1TOZ:A), (P01135, 3TGF:A), (P00749, 1URK:A), (O14944, 1K36:A), (Q02763, 2GY5:A)
EGF_2	(P00736, 1APQ:A), (P01133, 1IVO:C), (P01132, 1EGF:A), (P00743, 1KIG:H), (P00741, 1PFX:L), (P00740, 1EDM:B), (P16293, 1X7A:C), (Q14393, 1H30:A), (Q99075, 1XDT:R), (Q60675, 1DYK:A), (P16581, 1G1T:A), (P16109, 1G1R:A), (O00187, 1SZB:A), (P46531, 1TOZ:A), (P07225, 1Z6C:A), (P01135, 1MOX:C), (P35442, 1YO8:A), (P35070, 1IOX:A), (O14944, 1K36:A), (Q02763, 2GY5:A)
GLYCOSYL_HYDROL_F5	(P06565, 1E5J:A), (P07985, 1CEC:A), (P19424, 1G01:A), (P23776, 1H4P:A)
GLYCOSYL_HYDROL_F10	(P07986, 1EXP:A), (P56588, 1BG4:A), (P26514, 1EOW:A)
HIPIP	(P00260, 1B0Y:A), (P59860, 3HIP:A)
HMA_1	(P35670, 2EW9:A), (P73241, 2GCF:A), (Q48271, 1YG0:A), (Q60048, 2AJ0:A)

IG_MHC	(P30443, 1W72:A), (P05534, 2BCK:A), (P10316, 1AQD:F), (P18464, 1E27:A), (P61770, 2BVO:B), (P30504, 1IM9:A), (P28078, 1K8I:A), (P01903, 1A6A:A), (P01888, 1BMG:A), (P61771, 1C16:_), (P01887, 1BII:B), (P61770, 2AV1:B), (P01864, 1BOG:B), (P01865, 1EGJ:H), (P01867, 1CIC:B), (P01904, 1KTD:A), (P01909, 1S9V:A), (P04225, 1JK8:A), (P14483, 1LNU:B), (P04228, 1IAO:A), (P01910, 1D9K:C), (P06344, 1K2D:B), (P01876, 1OWO:A), (P01861, 1ADQ:A), (P01837, 1AHW:A), (P01843, 1JNH:A), (P01842, 1JVK:A), (P25311, 1T7V:A), (P01880, 1ZVO:C)
IMP_1	(O14732, 2FVZ:A), (P20456, 2BJI:A), (Q9Z1N4, 1JP4:A)
KAZAL	(O96790, 1KMA:A), (P00995, 1CGI:A), (P01001, 2BUS:A), (P01003, 1OVO:A), (P67954, 1IY5:A), (Q9NQ38, 1HDL:A), (P68436, 1M8B:A)
LIPASE_SER	(O42807, 1USW:A), (P00591, 1ETH:A), (P06857, 1RP1:A), (P16233, 1LPA:A), (P22088, 1HQD:A), (P26876, 1EX9:A), (P29183, 1HPL:A), (P61870, 1TIA:A), (P61871, 1LGY:A), (P61872, 1TIC:A), (P80035, 1K8Q:A)
PEROXIDASE_2	(P48534, 1APX:A), (O59651, 1ITK:A), (Q08129, 1SJ2:A), (P49012, 1LGA:A), (P11542, 1QPA:A), (P06181, 1B80:A), (Q02567, 1MN1:A), (P22195, 1SCH:A), (Q42578, 1QO4:_), (Q39034, 1QGJ:A), (Q42578, 1PA2:A), (P28314, 1LY8:A)
PROTEIN_KINASE_ST	(O05871, 1RWI:A), (O08679, 1ZMU:A), (O43293, 1YRP:A), (O43318, 2EVA:A), (O75582, 1VZO:A), (O75716, 2BUJ:A), (O94804, 2J7T:A), (O96017, 2CN5:A), (P00517, 1Q24:A), (P05771, 2IOE:A), (P06244, 1FOT:A), (P06782, 2EUE:A), (P06784, 2B9H:C), (P0A5S5, 1MRU:A), (P11309, 1XQZ:A), (P15056, 1UWH:A), (P15442, 1ZY4:A), (P16892, 2F49:A), (P19525, 2A19:B), (P30291, 1X8B:A), (P36507, 1S9I:A), (P36887, 1CTP:E), (P41743, 1ZRZ:A), (P43250, 2ACX:A), (P45983, 1UKH:A), (P49759, 1Z57:A), (P49761, 2EU9:A), (P50613, 1UA2:A), (P51955, 2JAV:A), (P53779, 1PMV:A), (P54646, 2H6D:A), (P63086, 1ERK:A), (P68400, 1JWH:A), (P78368, 2C47:A), (Q00534, 1G43N:A), (Q00535, 1H4L:A), (Q02750, 1S9J:A), (Q04759, 1XJD:A), (Q06486, 1CKI:A), (Q07785, 1OB3:A), (Q16659, 2I6L:A), (Q28021, 2F2U:A), (Q96SB4, 1WBP:A), (Q99683, 2CLQ:A), (Q9H2G2, 2J51:A), (Q9HBH9, 2AC5:A), (Q9HCP0, 2CMW:A), (Q9JIH7, 1T4H:A), (Q9JLS3, 1U5Q:A), (Q9P0L2, 2HAK:A), (Q9P1W9, 2IWI:A), (Q9UIK4, 1WMK:A), (Q9Y6M4, 2CHL:A)
SHIGA_RICIN	(P06750, 1RZO:A), (P11140, 1ABR:B), (P16094, 1AHA:A), (P22851, 1NIO:A), (P23339, 1GIK:A), (P24476, 1LP8:A), (P24817, 1CF5:A), (P33185, 1BRY:Y), (Q03464, 1APA:A), (Q40772, 1LLN:A), (Q9M6E9, 2Q3N:A)

SMALL_CYTOKINES_CC	(O89093, 1HA6:A), (P10147, 1B50:A), (P13236, 1HUM:A), (P51671, 1EOT:A), (P55773, 1G91:A), (P80075, 1ESR:A), (P80098, 1BO0:A), (Q16663, 2HCC:A), (Q92583, 1NR2:A)
SNAKE_TOXIN	(P01379, 1LSI:A), (P01382, 1NTN:A), (P01386, 1TXA:A), (P01391, 1YI5:A), (P01398, 1KBA:A), (P01414, 1TFS:A), (P01416, 1NTX:A), (P01426, 1IQ9:A), (P01427, 1NOR:A), (P01441, 1CB9:A), (P01442, 1CRE:A), (P01443, 1KBS:A), (P01451, 1RL5:A), (P01452, 1CDT:A), (P01467, 2CCX:A), (P07525, 1CHV:S), (P0C1Z0, 1B41:B), (P59276, 1JE9:A), (P60301, 1H0J:A), (P60304, 2CDX:A), (P60616, 1HC9:A), (P60770, 1V6P:A), (P60775, 1QKD:A), (P62375, 1KXI:A), (P80245, 1UG4:A), (P80958, 1ONJ:A), (P82849, 1G6M:A), (P83346, 1VYC:A), (Q98959, 1I02:A)
SUBTILASE_ASP	(P15926, 1XF1:A), (P27693, 1AH2:A), (P23188, 1P8J:A), (Q99405, 1WSD:A), (P00782, 1DUI:A), (P00780, 1SBC:A), (P07518, 1MEE:A), (P04189, 1SCJ:A), (P04072, 1TEC:E), (P29599, 1ST3:A), (P00781, 1BH6:A), (Q45670, 1DBI:A)
THIOL_PROTEASE_ASN	(O46427, 1NB3:A), (O60911, 1FH0:A), (O65039, 1S4V:A), (P00784, 1BP4:A), (P00785, 1AEC:A), (P00787, 1CPJ:A), (P05994, 1GEC:E), (P07688, 1ITO:A), (P08176, 1XKG:A), (P10056, 1MEG:A), (P14080, 1YAL:A), (P25250, 2F05:A), (P25774, 1GLO:A), (P43235, 1ATK:A), (P43236, 2F7D:A), (P53634, 1K3B:A), (P80067, 1JQP:A), (P83654, 1O0E:A)
THIOREDOXIN	(O43396, 1GH2:A), (O77404, 1O73:A), (O95881, 1SEN:A), (P07591, 1FB6:A), (P08003, 2DJ1:A), (P0A616, 2I1U:A), (P0AA30, 2AJQ:B), (P0AEG5, 1U3A:A), (P0AEG6, 1EEJ:A), (P0AEG7, 1G0T:A), (P17967, 2B5E:A), (P20857, 1THX:A), (P22803, 2FA4:A), (P23400, 1DBY:A), (P29448, 1XFL:A), (P30101, 2DMM:A), (P35160, 1ST9:A), (P36655, 1VRS:A), (P52230, 1T00:A), (P55059, 2DJJ:A), (P58162, 1UC7:A), (P80028, 1EP7:A), (P80579, 1NSW:A), (Q15084, 1X5D:A), (Q7KQL8, 1SYR:A), (Q922R8, 2DML:A), (Q99757, 1W4V:A), (Q9H3N1, 1X5E:A), (Q9V429, 1XW9:A), (Q0D840, 1WMJ:A)

TRYPSIN_SER	(O00187, 1Q3X:A), (P00734, 1B7X:A), (P00735, 1UCY:A), (P00736, 1GPZ:A), (P00740, 1RFN:A), (P00741, 1PFX:L), (P00743, 1KIG:H), (P00747, 1BUI:A), (P00749, 1C5W:A), (P00750, 1BDA:A), (P00752, 1HIA:A), (P00756, 1SGF:G), (P00759, 1TON:A), (P00760, 1AQ7:A), (P00761, 1AKS:A), (P00763, 1J16:A), (P00766, 1AB9:A), (P00767, 1AFQ:A), (P00770, 3RP2:A), (P00771, 1AZZ:A), (P00775, 1OS8:A), (P00776, 1SGC:A), (P03951, 1XX9:A), (P03952, 2ANW:A), (P04070, 1AUT:C), (P05805, 1FON:A), (P05981, 1Z8G:A), (P06681, 2I6Q:A), (P06870, 1SPJ:A), (P07338, 1KDQ:A), (P07477, 1TRN:A), (P08001, 1FIZ:A), (P08246, 1B0F:A), (P08311, 1AU8:A), (P08419, 1BRU:P), (P09872, 2AIP:A), (P10144, 1FQ3:A), (P12544, 1OP8:A), (P16293, 1X7A:A), (P18291, 1FI8:A), (P20231, 1A0L:A), (P24158, 1FUJ:A), (P35031, 1BIT:A), (P41140, 2SFA:A), (P49863, 1MZA:A), (P80219, 1EUF:A), (P98072, 1EKB:A), (P98119, 1A5I:A), (Q04756, 1YBW:A), (Q07006, 1HPG:A), (Q61955, 1NPM:A), (Q7SIG3, 1ELT:A), (Q92876, 1GVL:A), (Q9GL10, 1FIW:A), (Q9I8X1, 1OPO:A), (Q9Y5K2, 2BDG:A)
TYR_PHOSPHATASE_1	(O60729, 1OHC:A), (O60942, 2C46:A), (P08575, 1YGR:A), (P15273, 1PA9:A), (P17706, 1L8K:A), (P18031, 1BZC:A), (P18052, 1P15:A), (P23467, 2AHS:A), (P23470, 2NLK:A), (P24656, 1YN9:A), (P26045, 2B49:A), (P28827, 1RPM:A), (P29350, 1GWZ:A), (P35236, 1ZC0:A), (P54829, 2BIJ:A), (Q05923, 1M3G:A), (Q12913, 2CFV:A), (Q13332, 2FH7:A), (Q13614, 1LW3:A), (Q15256, 2A8B:A), (Q15262, 2C7S:A), (Q15678, 2BZL:A), (Q16827, 2G59:A), (Q16849, 2I1Y:A), (Q8NEJ0, 2ESB:A), (Q9BVJ7, 2IMG:A), (Q9H1R2, 1YZ4:A), (Q9Y6W6, 1ZZW:A), (Q9ZVN4, 1XRI:A)
UBIQUITIN_CONJUGAT_1	(O00762, 1I7K:A), (O14933, 1WZV:A), (P06104, 1AYZ:A), (P15731, 1QCQ:A), (P21734, 1FZY:A), (P35129, 1Z2U:A), (P50623, 2GJD:A), (P51668, 2C4P:A), (P52478, 1Q34:A), (P60604, 2CYX:A), (P61077, 1X23:A), (P61081, 1Y8X:A), (P61085, 2BEP:A), (P61086, 1YLA:A), (P61088, 2C2V:A), (P62253, 2AWF:A), (P62837, 2ESK:A), (P63146, 1JAS:A), (P63279, 1A3S:A), (P63283, 1KPS:C), (Q02159, 2UCZ:A), (Q5VVX9, 1YRV:A), (Q96LR5, 1Y6L:A), (Q9NPD8, 1YH2:A)
ZINC_FINGER_C2H2_1	(Q922H1, 1WIR:A), (P49711, 1X6H:A), (O35615, 1SRK:A), (Q60980, 1P7A:A), (P08047, 1SP1:A), (P19544, 1XF7:A), (P08045, 1ZNF:A), (P47043, 1ZW8:A), (O43298, 2CSH:A), (P15822, 1BB0:A), (Q9NPA5, 1X5W:A), (P08048, 7ZNF:A), (O60281, 1X3C:A), (Q9ULJ3, 1WJP:A), (Q9H4T2, 2COT:A), (Q96JM2, 1X6F:A), (Q96ME7, 2CTD:A), (O00488, 1ZR9:A), (P17028, 1X6E:A), (Q63HK5, 2DMI:A), (Q9BU19, 2DLK:A), (Q07230, 2I3L:A)

Table A.4: PROSITE-derived false negative test set.

2FE2S_FERREDOXIN	(P07014, 1NEK:B), (P19915, 1FFU:A), (P19921, 1N5W:A), (P21149, 1L5P:A), (P29166, 1C4A:A), (Q56223, 2FUG:3)
4FE4S_FERREDOXIN	(Q8GC87, 1H0H:B), (P00210, 1FXR:A), (P10245, 1IQZ:A), (P29603, 1SIZ:A)
AA_TRANSFER_CLASS_1	(P95468, 1AY4:A), (P77806, 1U08:A)
ADH_SHORT	(O93868, 1H5Q:A), (P97852, 1GZ6:A), (Q16698, 1W6U:A), (Q8NBQ5, 1YB1:A), (Q9BY49, 1YXM:A), (Q9RPT1, 2B4Q:A)
ASP_PROTEASE	(P54958, 1YG9:A)
BETA_LACTAMASE_A	(P52663, 1BUE:A)
BPTI_KUNITZ_1	(P0C1X2, 1Y62:A), (P56409, 1TOC:R)
CHITINASE_18	(P30922, 1OWQ:A), (Q7YS85, 1SV8:A), (Q8SPQ0, 1ZBW:A), (P36222, 1HJV:A), (Q29411, 1XHG:A), (Q6TMG6, 1SR0:A), (P49347, 1CNV:A), (P36911, 2EBN:A), (P36913, 1EOM:A), (Q8L5C6, 1TE1:A)
CYTOCHROME_P450	(O31440, 1IZO:A), (Q16647, 2IAG:A), (Q9KIZ4, 1PKF:A)
C_TYPE_LECTIN_1	(O54709, 1HQ8:A), (O93426, 1UMR:A), (P20937, 1QO3:C), (P22030, 1FVU:B), (P26718, 1KCG:A), (P78380, 1YPO:A), (P81397, 1SB2:A), (Q07108, 1E87:A), (Q13241, 1B6E:A), (Q64329, 1P4L:D), (Q6QLQ4, 2BPD:A), (Q9PRS8, 1GZ2:A)
EF_HAND	(O70200, 1WY9:A), (O81223, 1V1F:A), (P25815, 1J55:A), (P33764, 1KSO:A), (P53141, 1M45:A), (P55008, 2G2B:A), (P80511, 1E8A:A), (P97352, 2CXJ:A), (Q7G188, 1PXY:B), (Q8LAS7, 1UHN:A), (Q99584, 1YUT:A), (Q99653, 2E30:A)
EGF_1	(P00736, 1APQ:A), (P09871, 1NZI:A), (Q9UHX3, 2BO2:A), (P01130, 1HJ7:A), (O00187, 1SZB:A), (Q9JJS8, 1NTO:A), (P05979, 1CQE:A), (Q05769, 1CVU:A), (P07204, 1DX5:A), (P35442, 1YO8:A)
EGF_2	(Q01594, 1LK9:A), (P09871, 1NZI:A), (Q9UHX3, 2BO2:A), (P05979, 1CQE:A), (Q05769, 1CVU:A)
GLYCOSYL_HYDROL_F5	(Q8WPJ2, 2COH:A)
GLYCOSYL_HYDROL_F10	(Q00177, 1TA3:B)
HMA_1	(O14618, 2CRL:A), (P40202, 1JK9:B)
IG_MHC	(P42081, 1NCN:A), (P22301, 1LK3:A), (P18464, 1E28:A), (P01868, 1AHW:B)

KAZAL	(P16895, 1Y1B:A), (P19883, 2BU0:C), (P21674, 1LR9:A)
PEROXIDASE_2	(P80025, 2IPS:A), (P05164, 1CXP:A)
PROTEIN_KINASE_ST	(O96013, 2CDZ:A), (P47811, 1LEW:A), (P53778, 1CM8:A), (P72001, 2H34:A), (Q16539, 1DI9:A), (Q8WZ42, 1TKI:A), (Q9NQU5, 2C30:A), (Q9NWZ3, 2NRU:A), (Q9P286, 2F57:A)
SHIGA_RICIN	(P81446, 1M2T:A), (P84786, 2B7U:A)
SMALL_CYTOKINES_CC	(O00175, 1EIG:A)
SNAKE_TOXIN	(P28375, 1DRS:A), (P81782, 1F94:A), (Q9YGJ0, 1MR6:A)
THIOL_PROTEASE_ASN	(O14815, 1ZIV:A), (P07384, 1ZCM:A), (P0C1S6, 1PXV:A), (P17655, 1KFU:L), (P81297, 1CV8:A), (P82474, 1CQD:A), (P84346, 2BDZ:A), (P97571, 1KXR:A), (Q01532, 1A6R:A), (Q07009, 1MDW:A), (Q13867, 1CB5:A), (Q9UBX1, 1M6D:A)
THIOREDOXIN	(P32557, 1BED:A), (P45111, 1T3B:A), (P77202, 1V57:A), (Q9BRA2, 1WOU:A), (Q9CQM5, 1V9W:A)
TRYPSIN_SER	(P00757, 1SGF:A), (P20160, 1AE5:A), (P35030, 1H4W:A), (P36368, 1A05:A), (Q91516, 1BQY:A)
TYR_PHOSPHATASE_1	(O75365, 1R6H:A), (Q12923, 1WCH:A), (Q16667, 1FPZ:A), (Q16828, 1MKP:A), (Q78EG7, 1X24:A), (Q93096, 1RXD:B), (Q9NRW4, 1WRM:A)
UBIQUITIN_CONJUGAT	(P25604, 1UZX:A), (P53152, 1JAT:B), (Q15819, 1J74:A), (Q8N2K1, 2F4W:B), (Q8WVN8, 1ZU0:A), (Q99816, 1KPP:A)
ZINC_FINGER_C2H2_1	(Q9VPQ6, 1FU9:A), (Q9UKY1, 2DJ0:A)

Table A.5: PROSITE-derived false positive test set.

2FE2S_FERREDOXIN	(Q02747, 1GNA:A)
ADH_SHORT	(O60547, 1T2A:A), (P0A114, 1GGV:A), (P0A115, 1DIN:A), (P0A7S9, 1P6G:M), (P0A7T1, 2GY9:M), (P0AC88, 1DB3:A), (P26391, 1G1A:A)
ASP_PROTEASE	(O29844, 2F6U:A), (P00808, 1I2S:A), (P11938, 1IGN:A), (P32797, 1KXL:A), (P37617, 1MWY:A), (Q9S5E2, 2D2X:A)
COPPER_BLUE	(P42230, 1Y1U:A), (P80256, 1DFE:A), (Q13133, 1UHL:B), (Q5SHR2, 1YL3:9), (Q92673, 2DM4:A), (Q9Z0Y9, 2ACL:B)
C_TYPE_LECTIN	(Q69ZL1, 1WGQ:A)

EF_HAND	(O00105, 1WMR:A), (O14561, 2DNW:A), (P12735, 1JJ2:C), (P69249, 1EJ7:S), (Q01745, 1GOF:A), (Q14764, 1Y7X:A), (Q8ZKF6, 1PG3:A)
EGF_1	(Q13241, 1B6E:A), (P06820, 1ING:A), (P05803, 1NCD:N)
EGF_2	(P56682, 1CCV:A), (Q12830, 2F6J:A), (P25777, 1FWO:A), (Q80TJ7, 1WEP:A), (P07174, 1SG1:X), (P19438, 1EXT:A), (P25942, 1CZZ:D), (P20334, 1D0J:G)
GLYCOSYL_HYDROL_F5	(Q8K1R3, 1WHU:A)
GLYCOSYL_HYDROL_F10	(O26249, 1L3B:A)
HMA_1	(P00118, 1F1F:A)
IG_MHC	(P11759, 1MFZ:A), (P78560, 3CRD:A), (P06149, 1F0X:A), (Q9UNA1, 1UGV:A), (Q8WZ42, 1G1C:A), (P17255, 1DFA:A)
LIPASE_SER	(O59893, 1BS9:A), (P07174, 1NGR:A), (P40363, 1PV1:A)
PEROXIDASE_2	(O75534, 1WFQ:A), (O43172, 1MZW:B)
SUBTILASE_ASP	(P77335, 1QOY:A), (P47712, 1BCI:A), (Q23229, 1MG7:A)
ZINC_FINGER_C2H2_1	(Q8MQJ9, 1Q7F:A), (P35555, 1APJ:A), (Q9HV00, 2FIY:A), (Q07WU7, 1E39:A), (P0C278, 1KSS:A), (Q8K310, 1X4D:A), (Q13133, 1UHL:B), (Q13888, 1Z60:A), (Q13049, 2CT2:A), (P15024, 1EJ6:B), (P07939, 2CSE:D), (Q9FKP8, 1WH5:A)

Table A.6: Patterns tested for sequence-based methods. We compared SeqFEATURE to three sequence-based methods on a subset of the PROSITE-derived test set, according to the procedure described in Section 3.1.3. We performed the comparison for Pfam in March 2007 and for Gene3D and Panther in October 2007, corresponding to version 21.0 for Pfam, 6.0 for Gene3D, and 6.1 for Panther.

Gene3D	Pfam	HMMPanther
2FE2S_FERREDOXIN	2FE2S_FERREDOXIN	AA_TRANSFER_CLASS_3
AA_TRANSFER_CLASS_3	4FE4S_FERREDOXIN	ADH_SHORT
ADH_SHORT	AA_TRANSFER_CLASS_3	ADH_ZINC
ADH_ZINC	ADH_SHORT	BPTI_KUNITZ_1
ADX	ADH_ZINC	CARBOXYLESTERASE_B_1
ASP_PROTEASE	ASP_PROTEASE	CHITINASE_18
BETA_LACTAMASE_B_1	BETA_LACTAMASE_A	CYTOCHROME_P450
BPTI_KUNITZ_1	BETA_LACTAMASE_B_1	IMP_1
CARBOXYLESTERASE_B_1	BPTI_KUNITZ_1	SMALL_CYTOKINES_CC
CHITINASE_18	CARBOXYLESTERASE_B_1	THIOL_PROTEASE_ASN
COPPER_BLUE	CHITINASE_18	TYR_PHOSPHATASE_1
C_TYPE_LECTIN	COPPER_BLUE	UBIQUITIN_CONJUGAT_1
CYTOCHROME_P450	CYTOCHROME_P450	
EF_HAND	C_TYPE_LECTIN_1	
GLYCOSYL_HYDROL_F5	EF_HAND	
HIPIP	GLYCOSYL_HYDROL_F5	
HMA_1	GLYCOSYL_HYDROL_F10	
IMP_1	HIPIP	
KAZAL	HMA_1	
LIPASE_SER	IMP_1	
PROTEIN_KINASE_ST	PEROXIDASE_2	
SHIGA_RICIN	PROTEIN_KINASE_ST	
SMALL_CYTOKINES_CC	SHIGA_RICIN	
SNAKE_TOXIN	SMALL_CYTOKINES_CC	
THIOL_PROTEASE_ASN	SNAKE_TOXIN	
THIOREDOXIN	THIOL_PROTEASE_ASN	
TRYPSIN_SER	THIOREDOXIN	
TYR_PHOSPHATASE_1	TRYPSIN_SER	
UBIQUITIN_CONJUGAT_1	TYR_PHOSPHATASE_1	
	UBIQUITIN_CONJUGAT_1	
	ZINC_FINGER_C2H2_1	

Table A.7: Test sets for structure-based method comparison.

Positive set		Negative set	
PROSITE Pattern	PDB ID	PROSITE Pattern	PDB ID
AA_TRANSFER_CLASS_1	1AMA	ADH_SHORT	1DB3
AA_TRANSFER_CLASS_1	1YAA	ADH_SHORT	1T2A:A
ADH_SHORT	1HDR	ADH_SHORT	1G1A
ADH_SHORT	1YDE	ADH_SHORT	1GGV:A
ADH_ZINC	1HSO	ADH_SHORT	1DIN
ADH_ZINC	1E3E	ASP_PROTEASE	1I2S
BETA_LACTAMASE_A	1N9B	ASP_PROTEASE	1IGN
BETA_LACTAMASE_B_1	1X8G	ASP_PROTEASE	1KXL
CARBOXYLESTERASE_B_1	1GZ7	ASP_PROTEASE	1MWY
CARBOXYLESTERASE_B_1	1K4Y	ASP_PROTEASE	2F6U
CHITINASE_18	1XHG	GLYCOSYL_HYDROL_F10	1F38
CHITINASE_18	1HJV	GLYCOSYL_HYDROL_F5	1WHU
CYTOCHROME_P450	2IAG	PEROXIDASE_2	1MZW
GLYCOSYL_HYDROL_F5	1H4P	PEROXIDASE_2	1WFQ
IMP_1	1JP4	PROTEIN_KINASE_ST	1LUF
IMP_1	2FVZ		
PEROXIDASE_2	1LGA		
PEROXIDASE_2	1LY8		
SHIGA_RICIN	1LLN		
SHIGA_RICIN	1LP8		
THIOL_PROTEASE_ASN	1OOE		
THIOL_PROTEASE_ASN	1CQD		
THIOREDOXIN	1EEJ		
THIOREDOXIN	1V9W		
THIOREDOXIN	2B5E		
THIOREDOXIN	1EP7		
TRYPSIN_SER	1KDQ		
UBIQUITIN_CONJUGAT_1	1WZV		
UBIQUITIN_CONJUGAT_1	1Y8X		

Appendix B

Predictions for TargetDB structures

This section contains predictions for structural genomics targets with unknown function registered in the TargetDB repository up to August 2008. All predictions listed scored higher than the 100% specificity cutoff for the named model, except for one case mentioned in the text of Section 3.2.4. The table lists the PDB ID of the solved structure, the model for the predicted functional site, the z-score of the predicted site hit, and the residue ID of the site hit. Where hits were identified in identical protein chains, only the highest scoring chain for each model or residue ID is shown.

Table B.1: Predictions for structural genomics targets with unknown function.

PDB ID	Model name	Z-score	Residue ID	Chain
1DFC	EF_HAND.3.ASN.ND2	3.136	ASN1077	A
1DQZ	LIPASE_SER.7.SER.OG	4.817	SER124	A
1F8I	LIPASE_SER.7.SER.OG	5.237	SER317	B
1F05	HMA_1.5.CYS.SG	3.629	CYS13	A
1IA1	RNASE_T2_1.4.HIS.ND1	3.982	HIS44	B
1IL0	HMA_1.5.CYS.SG	3.549	CYS11	A
1IW7	PROTEIN_KINASE_ST.5.ASP.OD2	4.129	ASP74	K
1J03	EF_HAND.9.THR.OG1	5.062	THR74	A
1J1Z	PROTEIN_KINASE_ST.5.ASP.OD2	4.259	ASP34	D

1J20	PROTEIN_KINASE_ST.5.ASP.OD2	4.279	ASP34	D
1J21	PROTEIN_KINASE_ST.5.ASP.OD2	4.318	ASP34	C
1J6U	EF_HAND.7.THR.OG1	3.599	THR110	A
1K1E	PROTEIN_KINASE_ST.5.ASP.OD2	4.137	ASP16	K
1K77	RNASE_T2_1.4.HIS.ND1	4.063	HIS202	A
1KH2	PROTEIN_KINASE_ST.5.ASP.OD2	4.423	ASP34	B
1KH3	PROTEIN_KINASE_ST.5.ASP.OD2	4.26	ASP34	B
1KOR	PROTEIN_KINASE_ST.5.ASP.OD2	4.54	ASP34	C
1M33	LIPASE_SER.7.SER.OG	4.681	SER82	A
1MRU	PROTEIN_KINASE_ST.5.ASP.OD2	4.448	ASP138	A
1NF2	EF_HAND.3.ASN.ND2	3.178	ASN716	C
1O5U	EF_HAND.7.THR.OG1	3.598	THR37	A
1ONO	RNASE_T2_1.4.HIS.ND1	4.2	HIS61	A
1OYZ	HMA_1.8.CYS.SG	2.955	CYS106	A
1PG6	EF_HAND.7.THR.OG1	3.8	THR206	A
1RVK	COPPER_BLUE.11.HIS.NE2	3.362	HIS24	A
1S9U	EF_HAND.12.TYR.OH	4.085	TYR175	A
1SFS	KAZAL.7.CYS.SG	2.372	CYS21	A
1SQ1	PROTEIN_KINASE_ST.5.ASP.OD2	4.139	ASP224	A
1SYR	THIOREDOXIN.11.CYS.SG	3.158	CYS41	C
1SYR	THIOREDOXIN.8.CYS.SG	3.03	CYS38	B
1TO3	HMA_1.5.CYS.SG	3.637	CS167	A
1TU9	EF_HAND.12.TYR.OH	4.071	TYR25	A
1UG2	HMA_1.5.CYS.SG	3.609	CYS88	A
1V5N	ZINC_FINGER_C2H2_1.3.CYS.SG	4.141	CYS53	A
1VKA	EF_HAND.12.TYR.OH	4.187	TYR44	A
1VLY	RNASE_T2_1.4.HIS.ND1	4.036	HIS62	A
1WIL	ZINC_FINGER_C2H2_1.3.CYS.SG	3.845	CYS21	A
1WJJ	EF_HAND.7.THR.OG1	3.753	THR72	A
1X5W	ZINC_FINGER_C2H2_1.3.CYS.SG	3.972	CYS15	A
1X6F	ZINC_FINGER_C2H2_1.3.CYS.SG	4.316	CYS31	A
1XHS	EF_HAND.7.THR.OG1	3.798	THR71	A
1XKQ	ADH_SHORT.3.TYR.OH	4.971	TYR162	B
1XRI	TYR_PHOSPHATASE_1.3.CYS.SG	7.553	CYS150	B
1Y12	EF_HAND.7.THR.OG1	3.933	THR20	A
1Y1X	EF_HAND.1.ASP.OD1	6.197	ASP37	B
1Y1X	EF_HAND.1.ASP.OD2	4.636	ASP37	A
1Y1X	EF_HAND.3.ASP.OD1	5.51	ASP39	A
1Y23	ZINC_FINGER_C2H2_1.1.CYS.SG	4.413	CYS7	C
1YDG	PROTEIN_KINASE_ST.5.ASP.OD2	4.136	ASP54	B
1YDW	PEROXIDASE_1.8.HIS.NE2	3.572	HIS133	A
1YI7	PROTEIN_KINASE_ST.5.ASP.OD2	4.116	ASP526	B
1YJE	HMA_1.8.CYS.SG	2.951	CYS550	A
1Z84	ZINC_FINGER_C2H2_1.1.CYS.SG	4.713	CYS63	B
1Z84	ZINC_FINGER_C2H2_1.9.HIS.ND1	2.847	HIS83	A
1ZKP	BETA_LACTAMASE_B_1.4.HIS.ND1	4.557	HIS59	B
1ZKP	BETA_LACTAMASE_B_1.4.HIS.NE2	6.458	HIS59	B
1ZKP	BETA_LACTAMASE_B_1.6.HIS.ND1	7.415	HIS61	C
1ZKP	BETA_LACTAMASE_B_1.8.ASP.OD1	6.952	ASP63	C
1ZR9	ZINC_FINGER_C2H2_1.1.CYS.SG	4.423	CYS45	A

2A20	EF_HAND.12.TYR.OH	4.169	TYR58	F
2AZ4	BETA_LACTAMASE_B_1.4.HIS.NE2	6.448	HIS92	C
2AZ4	BETA_LACTAMASE_B_1.6.HIS.ND1	5.762	HIS94	B
2AZ4	BETA_LACTAMASE_B_1.6.HIS.NE2	5.632	HIS404	B
2AZ4	BETA_LACTAMASE_B_1.8.ASP.OD1	6.086	ASP96	B
2B67	PROTEIN_KINASE_ST.5.ASP.OD2	4.458	ASP126	A
2COT	ZINC_FINGER_C2H2_1.1.CYS.SG	4.516	CYS49	A
2CQ7	KAZAL.1.CYS.SG	1.873	CYS27	A
2CUQ	ZINC_FINGER_C2H2_1.1.CYS.SG	4.444	CYS18	A
2D9H	ZINC_FINGER_C2H2_1.1.CYS.SG	4.415	CYS10	A
2DCL	EF_HAND.3.ASN.ND2	3.221	ASN19	A
2DIP	ZINC_FINGER_C2H2_1.3.CYS.SG	4.218	CYS52	A
2E72	ZINC_FINGER_C2H2_1.3.CYS.SG	2.796	CYS380	A
2EJQ	ZINC_PROTEASE.4.GLU.OE1	4.547	GLU96	A
2EQ0	ZINC_FINGER_C2H2_1.1.CYS.SG	3.088	CYS459	A
2EQ0	ZINC_FINGER_C2H2_1.1.CYS.SG	3.233	CYS414	A
2EQ0	ZINC_FINGER_C2H2_1.3.CYS.SG	2.915	CYS462	A
2EQ0	ZINC_FINGER_C2H2_1.3.CYS.SG	2.918	CYS490	A
2EQ0	ZINC_FINGER_C2H2_1.3.CYS.SG	3.049	CYS686	A
2EQ0	ZINC_FINGER_C2H2_1.3.CYS.SG	3.082	CYS417	A
2F9C	EF_HAND.7.THR.OG1	3.895	THR220	A
2FH7	TYR_PHOSPHATASE_1.3.CYS.SG	7.719	CYS1880	A
2FH7	TYR_PHOSPHATASE_1.3.CYS.SG	7.943	CYS1589	A
2G59	TYR_PHOSPHATASE_1.3.CYS.SG	7.776	CYS225	A
2GB3	PROTEIN_KINASE_ST.5.ASP.OD2	4.265	ASP2	D
2GLZ	ZINC_FINGER_C2H2_1.1.CYS.SG	4.381	CYS19	B
2HRZ	ADH_SHORT.3.TYR.OH	5.056	TYR159	A
2HY3	TYR_PHOSPHATASE_1.3.CYS.SG	7.231	CYS1060	B
2I1Y	TYR_PHOSPHATASE_1.3.CYS.SG	7.912	CYS909	B
2NVP	SHIGA_RICIN.5.GLU.OE1	4.131	GLU71	A
2OGF	EF_HAND.9.THR.OG1	4.675	THR17	D
2OX6	EF_HAND.9.ASN.OD1	4.102	ASN8	B
2P7H	ZINC_PROTEASE.4.GLU.OE1	3.657	GLU113	D
2POZ	AA_TRANSFER_CLASS_1.4.LYS.NZ	4.196	LYS157	B
2QRU	PROTEIN_KINASE_ST.5.ASP.OD1	3.82	ASP164	A
2QYE	AA_TRANSFER_CLASS_1.4.LYS.NZ	3.483	LYS165	C
2RD9	ZINC_FINGER_C2H2_1.9.HIS.ND1	2.893	HIS-7	A
2YS4	UBIQUITIN_CONJUGAT_1.10.CYS.SG	3.342	CYS64	A
3BIJ	EF_HAND.9.ASN.ND2	4.218	ASN93	A
3BJQ	ZINC_PROTEASE.4.GLU.OE1	3.774	GLU123	F
3C2Q	ZINC_PROTEASE.4.GLU.OE1	3.666	GLU249	A
3CE2	ZINC_PROTEASE.4.GLU.OE1	4.534	GLU401	A
3CE2	ZINC_PROTEASE.4.GLU.OE2	4.635	GLU401	A
3CLW	GLYCOSYL_HYDROL_F5.7.GLU.OE2	5.174	GLU209	D
3D19	SHIGA_RICIN.5.GLU.OE2	3.881	GLU26	D
3D19	SHIGA_RICIN.5.GLU.OE2	3.693	GLU154	D

Appendix C

CYS clustering supplementary data

C.1 Zinc sub-cluster analysis

We combined zinc-binding sub-clusters from four coarse clusters and repeated the cluster selection process. The output corresponded almost exactly to the original sub-clusters (see Figure C.1), with the only exceptions being two microenvironments that became singletons. This indicates that the coarse k-means clustering is partitioning the microenvironments reasonably well.

We also examined the principal component vectors for sub-clusters representing the same type of zinc binding (e.g. C2H2, 4 CYS, etc). There are real differences between the microenvironments despite their binding zinc in the same fashion (see Figures C.2–C.5). Although principal components do not correspond directly to physicochemical properties, we can map them back to their major constituent properties to get a more intuitive understanding of the microenvironment differences.

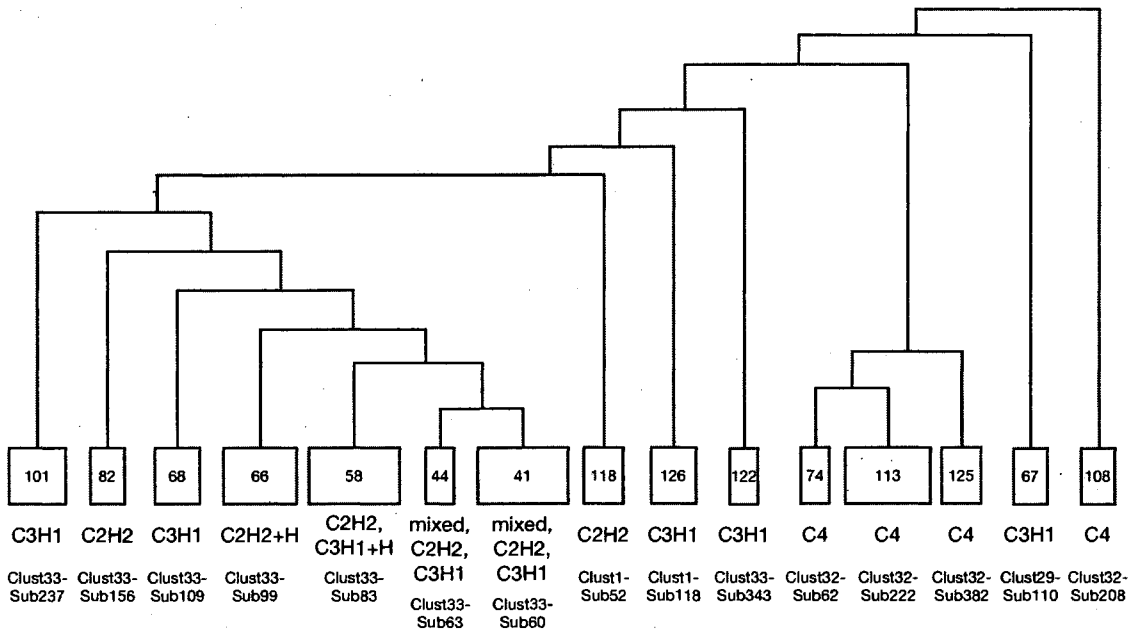


Figure C.1: Hierarchical tree from combined zinc sub-cluster analysis. Each sub-cluster is shown as a yellow box, with the width proportional to the size of the sub-cluster. We label each sub-cluster with the new node name inside the box, and the original sub-cluster ID below. We also indicate the type of zinc-binding represented by the sub-cluster.

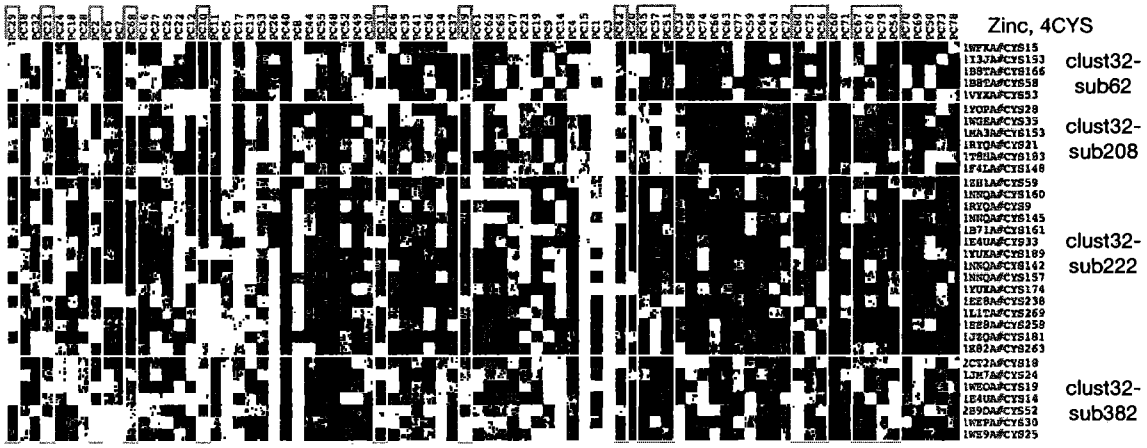


Figure C.2: Comparison of principal component vectors for sub-clusters binding zinc with 4 CYS.

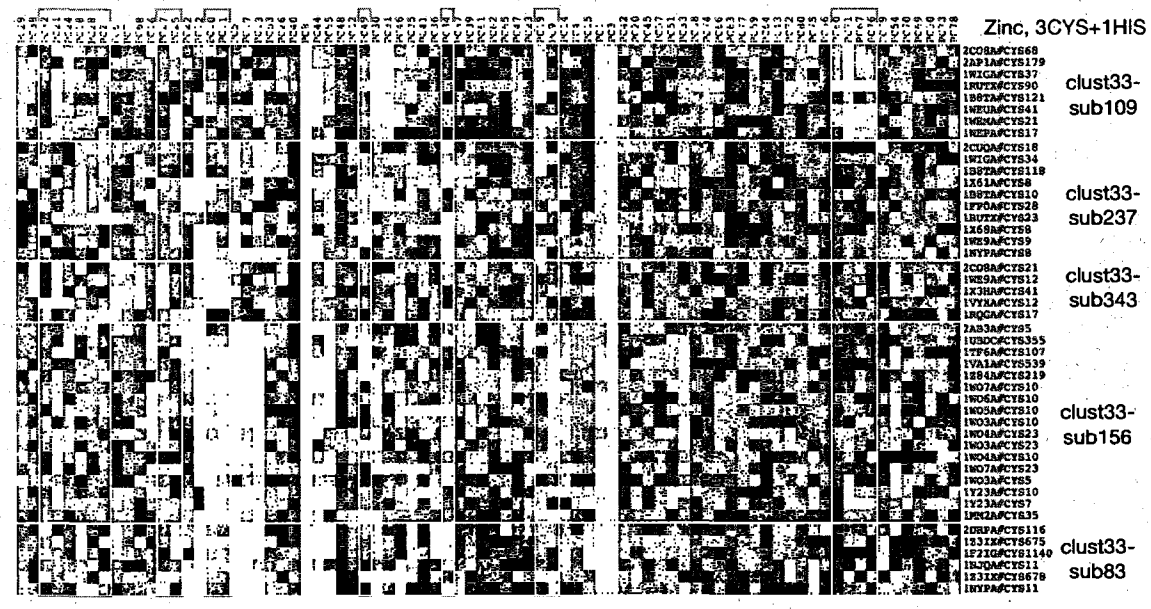


Figure C.3: Comparison of principal component vectors for sub-clusters binding zinc with 3 CYS and 1 HIS.

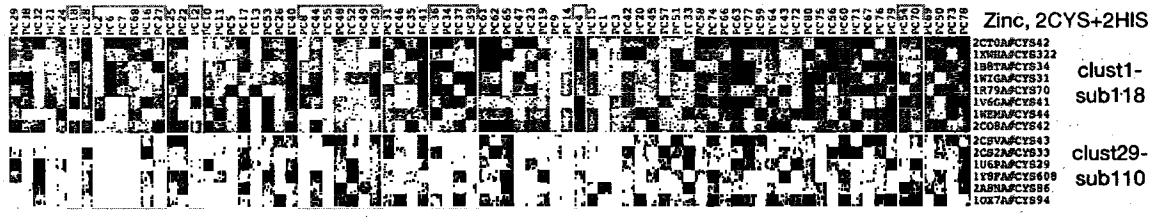


Figure C.4: Comparison of principal component vectors for sub-clusters binding zinc with 2 CYS and 2 HIS.

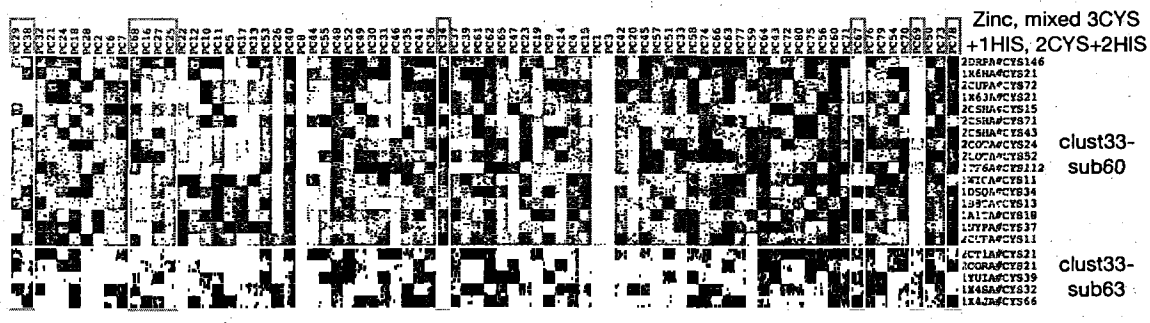


Figure C.5: Comparison of principal component vectors for sub-clusters binding zinc with either 3 CYS and 1 HIS or 2 CYS and 2 HIS.

C.2 Summary information for CYS sub-clusters

This section contains summary information for each sub-cluster produced from the 40 CYS-based k-means clusters using the methods described in Section 5.2.3. Detailed annotations are available online [167].

Sub-clusters for each coarse-grained cluster are contained in the same table. The sub-cluster ID (based on the node label from hierarchical clustering), list of PDB IDs, site residue IDs, and UniProt IDs are shown for each sub-cluster, in addition to a brief description of themes emerging from annotation and visual inspection of the environment. Note that only Swiss-Prot records are used to generate annotations, though the UniProt ID for each protein is listed here. If no table is present for a cluster, there were no sub-clusters returned by the cluster selection process containing at least 5 sites that had a functional coherence greater than 3.

Table C.1: Functionally coherent sub-clusters for Cluster 1

Potential annotation	Sub-cluster ID	PDB ID	Residue ID	UniProt ID
Zinc-binding, LIM domain, 3 CYS + 1 HIS coordination	NODE118X	2CUPA	CYS94	Q13642
		2CTOA	CYS42	Q8WV22
		1XWHA	CYS322	O43918
		1WIGA	CYS31	Q6H8Q1
		1R79A	CYS70	Q16760
		1V6GA	CYS41	Q6H8Q1
		1WEMA	CYS44	Q8C9B9
		2CO8A	CYS42	Q8TDZ2
		1B8TA	CYS142	P67966
			CYS34	
Copper-binding, multicopper oxidase proteins. 1 CYS + 2 HIS coordination.	NODE13X	1ZPUA	CYS484	P38993
		1GSKA	CYS492	P07788
		1V10A	CYS452	Q6H9H7
		1PF3A	CYS500	P36649
		1AOZA	CYS507	P37064
Associated with TYR phosphatases, adjacent to active site CYS	NODE257X	2FH7A	CYS1790	Q13332
		2B49A	CYS754	P26045
		2FH7A	CYS1501	Q13332
		2AHSA	CYS1812	P23467
		1BZCA	CYS121	P18031
		1YGRA	CYS1047	P08575
		2B30A	CYS361	P29350

Zinc-binding, C2H2 zinc finger/multi-HIS type, 1V7O:A = putative novel annotation	NODE52X	2BJ1A	CYS97	O58316
		1J6WA	CYS128	P44007
		1V7OA	CYS116	O58307
		1X6HA	CYS18	P49711
		1X6FA	CYS28	Q96JM2
		1K6YA	CYS43	P12497
		1WEEA	CYS19	Q9C810
Zinc-binding, 1 CYS + multi-HIS + ASP/GLU, di-nuclear zinc active sites, 1GY8:A/1UC2:A/1NYQ:A = putative novel annotations	NODE53X	2BI4A	CYS362	P0A9S1
		1UC2A	CYS98	O59245
		1T3IA	CYS372	Q55793
		1GY8A	CYS274	Q8T8E9
		1JF9A	CYS364	P77444
		1Z2WA	CYS41	Q9QZ88
		1TSRA	CYS182	P04637
		1BC2A	CYS168	P04190
		1XFJA	CYS114	Q9AAV3
		1T8HA	CYS125	P84138
		1RV9A	CYS118	Q9K0A8
		1NYQA	CYS181	Q8NW68
		1A7TA	CYS164	P25910

Table C.2: Functionally coherent sub-clusters for Cluster 2

Potential annotation	Sub-cluster ID	PDB ID	Residue ID	UniProt ID
Unknown function, possibly structural. Helix-based CYS surrounded by neutral, non-polar residues.	NODE14X	1XFKA	CYS319	Q9KSQ2
		1XDKB	CYS258	P22605
		1K2WA	CYS69	Q59787
		1U6GC	CYS233	Q86VP6
		1COJA	CYS114	Q9X6W9
Unknown function. Helical CYS with one or more sulfur-containing residues nearby.	NODE26X	2BX6A	CYS311	O75695
		1YJ5A	CYS408	Q9JLV6
		1XMIA	CYS590	P13569
		1VR9A	CYS23	Q9WZZ4
		1H6PA	CYS106	Q15554
		1DCFA	CYS43	P49333

Table C.3: Functionally coherent sub-clusters for Cluster 3

Potential annotation	Sub-cluster ID	PDB ID	Residue ID	UniProt ID
Unknown function, possibly structural. Mixed secondary structure CYS, often with a TYR nearby.	NODE93X	2NAPA	CYS282	P81186
		1KQFA	CYS380	P24183
		2FA1A	CYS144	P16684
		2C24A	CYS162	P71140
		1FFTA	CYS234	P0ABJ6
		1L5JA	CYS607	P36683
		2A3LA	CYS676	O80452
		1RXXA	CYS409	P13981
		1L7CA	CYS526	P35221

Table C.4: Functionally coherent sub-clusters for Cluster 4

Potential annotation	Sub-cluster ID	PDB ID	Residue ID	UniProt ID
Unknown function, probably structural. Extended beta sheet with $\geq$ 3 CYS flanked by PHE + ILE/LEU/VAL.	NODE23X	1GQ8A	CYS170	P83218
		2BX6A	CYS105	O75695
			CYS86	
		1DBGA	CYS203	Q46079
			CYS165	

Table C.5: Functionally coherent sub-clusters for Cluster 5

Potential annotation	Sub-cluster ID	PDB ID	Residue ID	UniProt ID
TYR kinase-associated site, potentially autocatalytic phosphorylation site (based on 1K9A:A). 3 sites are from unicellular enzymes, another is a viral coat protein. The environment is characterized by a loop-based central CYS with a nearby MET, and a TYR in the case of TYR kinases.	NODE70X	2B7AA	CYS1094	O60674
		1U46A	CYS356	Q07912
		1R0PA	CYS1308	P08581
		1MP8A	CYS647	Q05397
		1QCFA	CYS487	P08631
		1K9AA	CYS411	P32577
		1LUFA	CYS826	Q62838
		1XBAA	CYS597	P43405
		1Z45A	CYS155	P04397
		1BWDA	CYS264	P08078
		1AUYA	CYS132	P03608
		1J32A	CYS315	Q8RR70

Table C.6: Functionally coherent sub-clusters for Cluster 6

Potential annotation	Sub-cluster ID	PDB ID	Residue ID	UniProt ID
Unknown function, possibly ligand-binding. 4/5 sites adjacent to ligand(s). 1U6L:A near several selenomethionines.	NODE240X	2D1EA	CYS162	Q55891
		1U6LA	CYS15	Q02JJ4
		1ORRA	CYS10	P14169
		1U2ZA	CYS405	Q04089
		1TV5A	CYS233	Q08210

Table C.7: Functionally coherent sub-clusters for Cluster 7

Potential annotation	Sub-cluster ID	PDB ID	Residue ID	UniProt ID
Unknown function, potentially structural, possibly related to ligand-binding. 2 proteins are mitochondrial.	NODE287X	2APGA	CYS217	P95480
		1KCZA	CYS69	Q05514
		1DCIA	CYS181	Q62651
		1VPLA	CYS201	Q9WZ14
		1HZDA	CYS194	Q13825

Table C.8: Functionally coherent sub-clusters for Cluster 8

Potential annotation	Sub-cluster ID	PDB ID	Residue ID	UniProt ID
Unknown function, likely structural. Proteins cover a broad range of enzymatic and other functions. Environment is characterized by a helical CYS with an abundance of neutral, non-polar residues (specifically ILE, LEU, VAL) surrounding the sidechain.	NODE25X	2BUJA	CYS133	O75716
		1YQ7A	CYS152	P14324
		1H05A	CYS90	P0A4Z6
		1SNYA	CYS126	Q9W3H4
		1W66A	CYS117	Q10404
		1L1QA	CYS140	Q967M2
		1HU9A	CYS698	P09186
		1WA5B	CYS278	Q02821
		1Q1SC	CYS272	P52293
		2AVDA	CYS198	Q86VU5
		1VRWA	CYS249	Q9BH77
Unknown function, possibly structural. Usually helical CYS, always in the vicinity of HIS and multiple ILE/LEU/VAL.	NODE325X	2GSAA	CYS72	P24630
		2EW2A	CYS103	Q831Q5
		1GS0A	CYS222	O88554
		1SW6A	CYS473	P09959
			CYS354	
		1PXYA	CYS201	Q7G188

Table C.9: Functionally coherent sub-clusters for Cluster 10

Potential annotation	Sub-cluster ID	PDB ID	Residue ID	UniProt ID
Iron-binding (2FE2S) and zinc-binding (4 CYS). Presence of 4 sulfur atoms in each case.	NODE26X	1YQ3B	CYS70	Q9YHT2
		1KF6B	CYS62	P0AC47
		1T3QA	CYS53	P72223
			CYS56	
		1E7PB	CYS62	P17596
		1KWGA	CYS106	O69315
		1VD4A	CYS132	P29083
Unknown function. 2 proteins are growth factors, 2 are nucleotide-binding.	NODE162X	1IHKA	CYS134	P31371
		1G5HA	CYS308	Q9QZM2
		1IJTA	CYS155	P08620
		1X4NA	CYS46	Q91WJ8
		1A34A	CYS147	P17574
		1I2DA	CYS509	Q12650
		1RQ5A	CYS783	Q6RSN8

Table C.10: Functionally coherent sub-clusters for Cluster 11

Potential annotation	Sub-cluster ID	PDB ID	Residue ID	UniProt ID
Unknown function, possibly related to ligand-binding. Environment is usually helical and contains several of [ASP/GLU/ASN/GLN/ARG].	NODE33X	2FG5A	CYS120	Q13636
		1UKVY	CYS123	P01123
		1Z06A	CYS150	O35963
		1Z5VA	CYS392	P23258
		1TXUA	CYS225	Q9UJ41
		1UWCA	CYS235	O42807
		1A6ZA	CYS127	Q30201
Unknown function. Half are acyltransferases. Environment is mixed secondary structure with a PRO + several of [ASP/GLU/ASN/ARG].	NODE48X	1M6EX	CYS287	Q9SPV4
		1E1HA	CYS133	Q45894
		1DQ8A	CYS827	P04035
		1G0DA	CYS138	P52181
		1EVUA	CYS188	P00488
		1KV3A	CYS143	P21980
Unknown function, possibly related to ligand-binding. 6/15 proteins are methyltransferases which bind S-adenosyl-L-methionine. Others bind other ligands. The environment is usually adjacent to the ligand.	NODE136X	2FEAA	CYS121	O31667
		1Q0SA	CYS33	P04392
		2F8LA	CYS127	Q71Z85
		1VE3A	CYS47	O57965
		1RI1A	CYS73	Q8SR66
		1R74A	CYS65	Q14749
		1Y8CA	CYS47	Q97GJ5
		1P91A	CYS94	P36999
		1O9GA	CYS60	Q9F5K5
		1IM8A	CYS64	P43985
		1F38A	CYS1142	O26249
		1K9YA	CYS159	P32179
		1V8BA	CYS52	P50250
		1A7AA	CYS53	P23526
		1VJUA	CYS176	P84155

Table C.11: Functionally coherent sub-clusters for Cluster 15

Potential annotation	Sub-cluster ID	PDB ID	Residue ID	UniProt ID
Unknown function. All proteins are enzymes. Environments characterized by multiple ARG residues and occasionally a HIS.	NODE152X	2C46A	CYS91	O60942
		2C35A	CYS104	O15514
		1ZY9A	CYS68	O33835
		1YQQA	CYS227	P45563
		1TCUA	CYS233	Q9BMI9
		1FXUA	CYS231	P55859
Unknown function. Environments characterized by the presence of a MET, an ARG, and a PHE residue.	NODE279X	1YIRA	CYS350	Q9HW26
		1XCAA	CYS130	P29373
		1P4EA	CYS189	P03870
		1N52A	CYS409	Q09161
		1H2TC	CYS409	Q09161
		1GCZA	CYS56	P14174
		1DPTA	CYS56	P30046

Table C.12: Functionally coherent sub-clusters for Cluster 16

Potential annotation	Sub-cluster ID	PDB ID	Residue ID	UniProt ID
Unknown function. Environment characterized by multiple occurrences of [TYR, LYS, ARG, GLU], but otherwise somewhat sparse. Sub-cluster contains two uncharacterized SG proteins.	NODE42X	2A7OA	CYS72	Q9BYW2
		1ZXEA	CYS791	P15442
		1JSWA	CYS89	P0AC40
		1RYUA	CYS106	O14497
		1ME7A	CYS221	P50097
		1Z85A	CYS130	Q9X1A0
		1XJ5A	CYS104	Q9ZUB3
		1QYIA	CYS18	Q8NW41
Unknown function. A mixture of enzymes and DNA-binding or sugar-binding proteins.	NODE78X	2AF7A	CYS85	O26336
		1OB8A	CYS53	Q97YX6
		1J2ZA	CYS245	O25927
		1PHZA	CYS334	P04176
		1T5OA	CYS29	O29877
		1OF3A	CYS75	Q9RIK9
		1E1CB	CYS58	P11652
		1JMSA	CYS188	P09838

Table C.13: Functionally coherent sub-clusters for Cluster 18

Potential annotation	Sub-cluster ID	PDB ID	Residue ID	UniProt ID
Unknown function. Several proteins are nucleotide- or protein-binding and are oncogenes. Environment characterized by multiple TYR and PHE residues.	NODE134X	1YAFA	CYS135	P25052
		1OQ1A	CYS75	O31524
		1Q32A	CYS535	P38319
		1VPRA	CYS929	O77206
		1MR1C	CYS224	P12755
		1PYOB	CYS219	P42575
		1J5WA	CYS239	Q9WY59
		1K4SA	CYS300	P11387
Unknown function. Environment generally consists of helical central CYS with a TYR and often a GLN or ASN. Many proteins are enzymes or protein binding.	NODE22X	2A7LA	CYS91	Q96B02
		1Q1SC	CYS419	P52293
		1HNOA	CYS190	Q05871
		1EEMA	CYS112	P78417
		1FVAA	CYS107	P54149
		1ELWA	CYS62	P31948
		1YUEA	CYS255	P19896
		1WA5B	CYS214	Q02821
		1LC5A	CYS285	P97084
		1ELQA	CYS306	Q9ZHG9
		1LTUA	CYS52	P30967
		1U6GC	CYS506	Q86VP6
		1YODA	CYS36	

Table C.14: Functionally coherent sub-clusters for Cluster 19

Potential annotation	Sub-cluster ID	PDB ID	Residue ID	UniProt ID
Unknown function, possibly structural. Central CYS is inward facing on surface-exposed helix. 5 proteins are known to be phosphorylated.	NODE16X	2CW9A	CYS319	O43615
		1RT8A	CYS149	O59945
		2C46A	CYS110	O60942
		1VGYA	CYS113	Q9JYL2
		1SV0A	CYS66	Q01842
		1OMWA	CYS72	P21146
		1WEXA	CYS67	Q921F4
		1ZB6A	CYS209	Q4R2T2
		1E20A	CYS39	Q9SWE5
		1XTPA	CYS183	Q4Q7M2
		1E15A	CYS127	Q54276
		1BYWA	CYS108	Q12809
		1WLZA	CYS274	Q5THR3

Table C.15: Functionally coherent sub-clusters for Cluster 20

Potential annotation	Sub-cluster ID	PDB ID	Residue ID	UniProt ID
Unknown function, possibly enzyme-related. Central CYS on surface-exposed helix with a LYS sidechain nearby.	NODE83X	1XT9A	CYS64	Q96LD8
		1XSAA	CYS128	P50583
		1FPZA	CYS39	Q16667
		1T3QC	CYS89	P72222
		1D2ZB	CYS53	P22812

Table C.16: Functionally coherent sub-clusters for Cluster 21

Potential annotation	Sub-cluster ID	PDB ID	Residue ID	UniProt ID
TYR protein phosphatase active site environment. Mixed receptor and non-receptor type.	NODE5X	2AHSA	CYS1904	P23467
		2BZLA	CYS1121	Q15678
		2B49A	CYS842	P26045
		2FH7A	CYS1589	Q13332
			CYS1880	
Iron-binding site, 2FE2S type with additional CYS. Somewhat sparse, strand environment.	NODE17X	1T3QA	CYS107	P72223
		1RM6C	CYS100	O33818
		1FO4A	CYS113	P80457
		1DGJA	CYS100	Q9REC4
		1N5WA	CYS102	P19921
TYR protein phosphatase active site environment. Non-receptor and secreted types, dual specificity and multifunctional proteins.	NODE27X	2C46A	CYS126	O60942
		1ZC0A	CYS270	P35236
		1G4US	CYS481	P74873
		1OHCA	CYS314	O60729
		1YN9A	CYS119	P24656
		1XRIA	CYS150	Q9ZVN4
		1D5RA	CYS124	P60484

WD-repeat-associated environment. The environment is characterized by beta sheets and the presence of another CYS (sometimes belonging to an adjacent microenvironment from this sub-cluster).	NODE48X	1VYHC	CYS281	P63005
		1P22A	CYS475	Q9Y297
			CYS435	
			CYS272	
			CYS312	
		1K8KC	CYS13	Q58CQ2
			CYS101	
		1ERJA	CYS349	P16649
		1NR0A	CYS541	Q11176

Table C.17: Functionally coherent sub-clusters for Cluster 22

Potential annotation	Sub-cluster ID	PDB ID	Residue ID	UniProt ID
Iron-binding site, 4FE4S type. Usually a LYS and PRO nearby. 1OKG is not annotated as binding iron.	NODE159X	1OKGA	CYS278	Q7K9G0
		1B25A	CYS284	O93738
		1KQFB	CYS175	P0AAJ3
		1KQFA	CYS92	P24183
		1H0HA	CYS54	Q934F5

Table C.18: Functionally coherent sub-clusters for Cluster 23

Potential annotation	Sub-cluster ID	PDB ID	Residue ID	UniProt ID
Cytochrome C-associated adjacent heme C/heme binding sites. Strand or slight helical environment with 2 CYS and 2+ HIS.	NODE44X	1SP3A	CYS101	Q8E9W8
		1Q08A	CYS18	Q9Z4P0
		1M1PA	CYS18	Q8EDL6
		1D4CA	CYS18	P83223
		1OFWA	CYS287	Q9RN68
		1OAHA	CYS320	Q8VNU2
		1GWSA	CYS508	P24092
		1M1PA	CYS61	Q8EDL6
		1FS9A	CYS298	Q9S1E5
		1GU6A	CYS285	P0ABK9
Cytochrome C3 and higher molecular weight cytochrome-associated mixed heme C and heme-binding site. Strand or slight helical environment.	NODE46X	3CA0A	CYS39	P94690
		2BQ4A	CYS56	P94691
		1GY0A	CYS52	Q9R638
		1OFWA	CYS241	Q9RN68
		1GWSA	CYS462	P24092
		1GM4A	CYS46	Q9L915
		1GWSA	CYS350	P24092
		1GY0A	CYS80	Q9R638
		1GWSA	CYS150	P24092
		1EHJA	CYS29	P00137
		1OFWA	CYS50	Q9RN68
		1GM4A	CYS33	Q9L915
		1GY0A	CYS39	Q9R638
		1GM4A	CYS79	Q9L915
		1GWSA	CYS280	P24092
		1QN0A	CYS83	P00133
		1OFWA	CYS130	Q9RN68

Cytochrome C-associated adjacent heme C/heme binding sites. Helical environmental with 2 CYS and 2 HIS.	NODE80X	3CA0A	CYS82	P94690
		1OFWA	CYS267	Q9RN68
		1GWSA	CYS488	P24092
		1W7OA	CYS92	Q6XCI5
		1GWSA	CYS83	P24092
Cytochrome C-associated mixed heme C and heme binding site. Environment characterized by presence of [CYS/MET/LYS] and 1+ PRO, and only 1 HIS.	NODE83X	2BGVX	CYS18	Q00499
		1ZRTD	CYS37	P08501
		1DVVA	CYS15	P00099
		1QN2A	CYS17	Q7SIA4
		1JDLA	CYS18	P81154
		1C7MA	CYS17	P54820
		1ETPA	CYS17	Q52369

Table C.19: Functionally coherent sub-clusters for Cluster 24

Potential annotation	Sub-cluster ID	PDB ID	Residue ID	UniProt ID
Unknown function. Environment contains ASP and GLU and usually 1 or more LYS. *1VPJA is now 2ISBA.	NODE17X	2A2CA	CYS303	Q01415
		1X6VA	CYS78	O43252
		1VPJA*	CYS100	O29167
		1FJ1E	CYS84	P14013
		1N52A	CYS36	Q09161

Table C.20: Functionally coherent sub-clusters for Cluster 25

Potential annotation	Sub-cluster ID	PDB ID	Residue ID	UniProt ID
Sugar kinase-associated site. Not the active site. Beta-sheet environment with multiple sulfur-containing residues.	NODE19X	1T5AA	CYS49	P14618
		1PKLA	CYS25	Q27686
		1E0TA	CYS8	P0AD61
		1BG3A	CYS704	P05708
		1BG3A	CYS256	P05708

Table C.21: Functionally coherent sub-clusters for Cluster 27

Potential annotation	Sub-cluster ID	PDB ID	Residue ID	UniProt ID
Unknown function. 2 have beta-sheets, the other 2 have helices. 3/4 have a MET or CYS (one disulfide).	NODE24X	1Z00B	CYS852	Q92889
		1TDPA	CYS34	P38582
		1JMSA	CYS404	P09838
		1WI0A	CYS159	P01730
			CYS130	

Table C.22: Functionally coherent sub-clusters for Cluster 28

Potential annotation	Sub-cluster ID	PDB ID	Residue ID	UniProt ID
Unknown function. Environment may be characterized by the presence of MET and several neutral, non-polar residues such as LEU, ILE, or VAL.	NODE122X	2AZEA	CYS314	Q14186
		1MDAH	CYS183	
			CYS167	
		1AORB	CYS317	P62871
		2AYNA	CYS104	P54578
		1Z6ZA	CYS130	P35270
		1QCOA	CYS315	P35505
Unknown function.	NODE178X	2B3HA	CYS340	P53582
		1YJ8A	CYS277	Q8I5P5
		1GWNA	CYS177	P61588
		1HUXA	CYS17	P11568
		1W6JA	CYS636	P48449
		1JXQA	CYS285	P55211

Table C.23: Functionally coherent sub-clusters for Cluster 29

Potential annotation	Sub-cluster ID	PDB ID	Residue ID	UniProt ID
Zinc binding site with 3 CYS + 1 HIS coordination. 1OX7 is a dinuclear site; 1OX7 and 2A8N are coordinated by just 2 CYS but have a nearby MET.	NODE110X	2CSVA	CYS43	Q14134
		2CS2A	CYS33	P09874
		1U6PA	CYS29	P03332
		1Y8FA	CYS608	Q62768
		2A8NA	CYS86	A9CK16
		1OX7A	CYS94	Q12178
Unknown function.	NODE113X	2AYVA	CYS85	
		1DUPA	CYS36	P06968
		1Y65A	CYS273	Q9KU37
		1WF6A	CYS49	Q92547
		1AORB	CYS271	P62871

Table C.24: Functionally coherent sub-clusters for Cluster 30

Potential annotation	Sub-cluster ID	PDB ID	Residue ID	UniProt ID
Iron binding site, oxidoreductase 2FE2S type. Coordination by 3-4 CYS. When 3 CYS, a MET is usually nearby. Little secondary structure.	NODE15X	1T3QA	CYS142	P72223
		1N5WA	CYS137	P19921
		1FO4A	CYS148	P80457
		1DGJA	CYS137	Q9REC4
		1JROA	CYS134	O54050
		1RM6C	CYS135	O33818
Iron binding site, oxidoreductase 2FE2S type. Coordination by 4 CYS. Sometimes an additional MET/CYS nearby. Some helical structure.	NODE24X	1T3QA	CYS110	P72223
		1N5WA	CYS105	P19921
		1RM6C	CYS103	O33818
		1JROA	CYS106	O54050
		1FO4A	CYS116	P80457
		1DGJA	CYS103	Q9REC4

Iron binding site, ferredoxin 2FE2S type. Coordination by 3-4 CYS. Little secondary structure.	NODE57X	1OQQA	CYS86	P00259
		1KRHA	CYS83	P07771
		1E9MA	CYS86	P80306
		1E6EB	CYS92	P00257
		1I7HA	CYS87	P0A9R4
Iron binding site, ferredoxin 2FE2S type. Coordination by 4 CYS with occasional MET/CYS nearby. Little secondary structure.	NODE110X	1OQQA	CYS48	P00259
		1KRHA	CYS49	P07771
		1E9MA	CYS48	P80306
		1I7HA	CYS51	P0A9R4
		1CZPA	CYS49	P0A3C8
		1E0ZA	CYS71	P00216
Iron binding site, ferredoxin 2FE2S type. Coordination by 4 CYS with CYS/MET nearby. Little secondary structure.	NODE122X	1OQQA	CYS45	P00259
		1E6EB	CYS52	P00257
		1E6EB	CYS46	P00257
		1KRHA	CYS46	P07771
		1I7HA	CYS48	P0A9R4
Iron binding site, oxidoreductases. Mixed 4FE4S and 2FE2S. Coordination by 4 CYS with 1 or 2 CYS/MET nearby. Little secondary structure.	NODE160X	1T3QA	CYS144	P72223
		1FO4A	CYS150	P80457
		1RM6C	CYS137	O33818
		1DGJA	CYS139	Q9REC4
		1JROA	CYS136	O54050
		1N5WA	CYS139	P19921
		1HFEL	CYS378	P07598
		1C4AA	CYS499	P29166
		1H2AS	CYS114	P21853
		1CC1S	CYS126	P13063

Table C.25: Functionally coherent sub-clusters for Cluster 31

Potential annotation	Sub-cluster ID	PDB ID	Residue ID	UniProt ID
SER/THR protein kinase-associated site. Located in domain IX in catalytic domain near substrate recognition site.	NODE14X	2EXEA	CYS362	P49761
		1VYWA	CYS191	P24941
		1BL6A	CYS211	Q16539
		1UKHA	CYS213	P45983
		1Q3DA	CYS245	P49841
		1BI7A	CYS207	Q00534
		1WBPA	CYS539	Q96SB4
		1HOWA	CYS592	Q03656
		1OKYA	CYS270	O15530
Unknown function, possibly protein-binding. Environment characterized by helical CYS with opposing TRP.	NODE18X	2COEA	CYS99	P04053
		1IMOA	CYS912	P49916
		1WF6A	CYS112	Q92547
		1WLMA	CYS46	Q9CRB6
		1IJAA	CYS126	Q9S446

Table C.26: Functionally coherent sub-clusters for Cluster 32

Potential annotation	Sub-cluster ID	PDB ID	Residue ID	UniProt ID
Strand-based, multi-nuclear zinc binding site. 1ML9 and 1MVH are tri-nuclear coordinated by 7 CYS. 1JJD is four-zinc site coordinated by 9 CYS and 2 HIS.	NODE46X	1MVHA	CYS307	O60016
		1ML9A	CYS128	Q8X225
		1JJDA	CYS54	P30331
			CYS47	
			CYS32	
			CYS16	
			CYS36	
Strand-based, single zinc binding site coordinated by 4 CYS. Environment is relatively sparse.	NODE62X	1WFKA	CYS15	Q9DAZ9
		1I3JA	CYS153	P13299
		1B8TA	CYS166	P67966
		1B8TA	CYS58	P67966
		1VYXA	CYS53	P90495
Strand-based, single zinc binding site coordinated by 4 CYS.	NODE208X	1YOPA	CYS28	Q3E840
		1WGEA	CYS35	Q8K0W9
		1MA3A	CYS153	O30124
		1RYQA	CYS21	Q8U440
		1T8HA	CYS183	P84138
		1F4LA	CYS148	P00959
Metal-binding site with 4 CYS coordination and little secondary structure. Site is mononuclear and usually a zinc, though 1YUX and 1B71 are iron-binding.	NODE222X	1ZH1A	CYS59	Q9WMX2
		1NNQA	CYS160	Q9UWP7
			CYS145	
			CYS142	
			CYS157	
		1RYQA	CYS9	Q8U440
		1B71A	CYS161	P24931
		1E4UA	CYS33	O95628
		1YUXA	CYS189	P30820
			CYS174	
		1EE8A	CYS238	O50606
			CYS258	
		1L1TA	CYS269	P84131
		1JZQA	CYS181	P56690
		1K82A	CYS263	P05523
Strand-based, single zinc binding site coordinated by 4 CYS. Environment is not especially sparse, and usually contains at least one GLU and sometimes an ASP.	NODE382X	2CT2A	CYS18	Q13049
		1JM7A	CYS24	P38398
		1WEOA	CYS19	Q9SWW6
		1E4UA	CYS14	O95628
		2B9DA	CYS52	P06465
		1WEPA	CYS30	Q80TJ7
		1WE9A	CYS25	O81488

Table C.27: Functionally coherent sub-clusters for Cluster 33

Potential annotation	Sub-cluster ID	PDB ID	Residue ID	UniProt ID
Zinc binding site with 3 CYS + 1 HIS coordination.	NODE109X	2C08A	CYS68	Q8TDZ2
		2AP1A	CYS179	Q8ZPZ9
		1WIGA	CYS37	Q6H8Q1
		1RUTX	CYS90	P70662
		1B8TA	CYS121	P67966
		1WEUA	CYS41	Q8C0D7
		1WEMA	CYS21	Q8C9B9
		1WEPA	CYS17	Q80TJ7
Zinc binding site, C2H2 type.	NODE156X	2DRPA	CYS116	P17789
		1Z3IX	CYS675	Q7ZV09
			CYS678	
		1F2IG	CYS1140	P08046
		1NJQA	CYS11	Q38895
		1NYPA	CYS11	P48059
Zinc binding site coordinated by 3 CYS + 1 HIS (except for 1NYP).	NODE237X	2CUQA	CYS18	Q13643
		1WIGA	CYS34	Q6H8Q1
		1B8TA	CYS118	P67966
			CYS10	
		1X61A	CYS8	Q15654
		1FP0A	CYS28	Q13263
		1RUTX	CYS23	P70662
		1X68A	CYS8	Q5TD97
		1WE9A	CYS9	O81488
		1NYPA	CYS8	P48059
Zinc binding site coordinated by 3 CYS + 1 HIS.	NODE343X	2C08A	CYS21	Q8TDZ2
		1WE9A	CYS12	O81488
		1X3HA	CYS41	O60711
		1VYXA	CYS12	P90495
		1RQGA	CYS176	Q9V011
Copper binding site with 2 CYS + 2 HIS coordination. All are blue copper type except for 1AQ8.	NODE49X	1PY0A	CYS78	P04377
		1OV8A	CYS122	P27197
		1ID2A	CYS93	P22365
		1BAWA	CYS89	Q51883
		1B3IA	CYS82	P50057
		1AQ8A	CYS136	P38501
Zinc binding site with either 2 CYS + 2 HIS or 3 CYS + 1 HIS coordination.	NODE60X	2DRPA	CYS146	P17789
		1X6HA	CYS21	P49711
		2CUPA	CYS72	Q13642
			CYS11	
		1X63A	CYS21	Q13642
		2CSHA	CYS15	O43298
			CYS71	
			CYS43	
		2C0TA	CYS24	Q9H4T2
			CYS52	
		1TF6A	CYS112	P03001
		1WIGA	CYS11	Q6H8Q1
		1DSQA	CYS34	P11284
		1B8TA	CYS13	P67966

		1A1TA	CYS18	Q75677
		1NYPA	CYS37	P48059
Zinc binding site with either 2 CYS + 2 HIS or 3 CYS + 1 HIS. Sparse environment.	NODE63X	2CT1A	CYS21	P49711
		2CORA	CYS21	P48059
		1YUIA	CYS39	Q08605
		1X4SA	CYS32	Q9UHR6
		1X4JA	CYS66	Q9H0F5
Zinc binding site, C2H2 type. A few sites have 3 CYS + 1 HIS coordination indicated in the structure, but in all cases there is an additional HIS within range that could potentially coordinate the zinc ion.	NODE83X	2AB3A	CYS5	
		1UBDC	CYS355	P25490
		1TF6A	CYS107	P03001
		1VA1A	CYS539	P08047
		1Z84A	CYS219	Q9FK51
		1WO7A	CYS10	Q92793
		1WO6A	CYS10	Q92793
		1WO5A	CYS10	Q92793
		1WO3A	CYS10	Q92793
			CYS23	
			CYS5	
		1WO4A	CYS23	Q92793
			CYS10	
		1WO7A	CYS23	Q92793
		1Y23A	CYS10	O07513
			CYS7	
		1MM2A	CYS35	Q14839
Metal-binding site with 3 or more HIS in the environment. Zinc binding is usually of C2H2 type with additional HIS nearby. Iron binding is with 1 CYS and 4 HIS. * 2JWO replaced 2A23.	NODE99X	2DRPA	CYS143	P17789
		2CT1A	CYS48	P49711
		1F2IG	CYS1107	P08046
		1UBDC	CYS298	P25490
		2CTDA	CYS65	Q96ME7
		2AMUA	CYS115	Q9WZC6
		1Y07A	CYS119	O83795
		1VZGA	CYS116	Q46495
		1GUPA	CYS55	P09148
		1GUPA	CYS52	P09148
		1Z84A	CYS216	Q9FK51
		2A23A*	CYS446	P21784
			CYS478	

Table C.28: Functionally coherent sub-clusters for Cluster 35

Potential annotation	Sub-cluster ID	PDB ID	Residue ID	UniProt ID
Unknown function. Environment is usually solvent exposed or in an unstructured part of the protein.	NODE181X	1T06A	CYS133	Q81BA8
		1M2OA	CYS69	P15303
		1SJJA	CYS42	P05094
		1EM6A	CYS495	P06737
		108UA	CYS74	Q93TU6

Table C.29: Functionally coherent sub-clusters for Cluster 36

Potential annotation	Sub-cluster ID	PDB ID	Residue ID	UniProt ID
Unknown function, potentially binding or catalytic. Solvent exposed environment with an ASP and LYS with the CYS.	NODE127X	1XM9A	CYS273	Q13835
		1FQVA	CYS123	Q13309
		1BG3A	CYS368	P05708
		1KPPA	CYS73	Q99816
		1AROP	CYS510	P00573

Table C.30: Functionally coherent sub-clusters for Cluster 39

Potential annotation	Sub-cluster ID	PDB ID	Residue ID	UniProt ID
Associated with viral proteins. Environment is sparse, with a TRP, ARG, TYR, and THR. 1TTU is an outlier.	NODE58X	2MEV1	CYS249	P12296
		1BBT1	CYS187	Q913V0
		1TTUA	CYS241	Q9TYY1
		1FPN1	CYS246	P04936
		1AL21	CYS270	P03300

Bibliography

[1] Alphey MS, WIlliams RAM, Mottram JC, Coombs GH, Hunter WN. (2003) The crystal structure of *Leishmania major* 3-mercaptopyruvate sulfurtransferase. *J Biol Chem* 278(48):48219-48227.

[2] Al-Shahrour F, Minguez P, Tarraga J, Medina I, Alloza E, Montaner D, Dopazo J. (2007) FatiGO+: a functional profiling tool for genomic data. Integration of functional annotation, regulatory motifs, and interaction data with microarray experiments. *Nucleic Acids Res* 35(Web Server issue):W91-W96.

[3] Altschul S, Gish W, Miller W, Myers E, Lipman D. (1990) Basic Local Alignment Search Tool. *J Mol Biol* 215:403-410.

[4] Altschul S, Madden T, Schaffer A, Zhang J, Zhang Z, Miller W, Lipman D. (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res* 25:3389-3402.

[5] Amigo E, Gonzalo J, Artiles J, Verdejo F. (2008) A comparison of extrinsic clustering evaluation metrics based on formal constraints. *Inf Retrieval* (published online July 28, 2008) doi: 10.1007/s10791-008-9066-8.

[6] Auld DS. (2001) Zinc coordination sphere in biochemical zinc sites. *BioMetals* 14:271-313.

[7] Ausiello G, Gherardini PF, Marcatili P, Tramontano A, Via A, Helmer-Citterich M. (2008) FunClust: a web server for the identification of structural motifs in a set of non-homologous protein structures. *BMC Bioinformatics* 9 Suppl 2:S2.

[8] Bagley SC, Altman RB. (1995) Characterizing the microenvironment surrounding protein sites. *Protein Sci* 4:622-635.

[9] Bagley SC, Wei L, Cheng C, Altman R. (1995) Characterizing oriented protein structural sites using biochemical properties. *Proc Int Conf Intell Syst Mol Biol* 12-20.

[10] Banatao DR, Altman RB, Klein TE. (2003) Microenvironment analysis and identification of magnesium binding sites in RNA. *Nucleic Acids Res* 31(15):4450-4460.

[11] Barrett AJ, Rawlings ND. (2001) Evolutionary lines of cysteine peptidases. *Biological Chemistry* 382(5):727-733.

[12] Bateman A, Haft DH. (2002) HMM-based databases in InterPro. *Briefings Bioinformatics* 3:236-244.

[13] Baumgartner Jr WA, Cohen KB, Fox LM, Acquaah-Mensah G, Hunter L. (2007) Manual curation is not sufficient for annotation of genomic databases. *Bioinformatics* 23:i41-i48.

[14] Becker KG, Hosack DA, Dennis Jr G, Lempicki RA, Bright TJ, Cheadle C, Engel J. (2003) PubMatrix: a tool for multiplex literature mining. *BMC Bioinformatics* 4:61.

[15] Berman HM, Westbrook J, Feng Z, Gilliland G, Bhat TN, Weissig H, Shindyalov IN, Bourne PE. (2000) The Protein Data Bank. *Nucleic Acids Res* 28(1):235-242.

[16] Binkowski TA, Naghibzadeg S, Liang J. (2003) CASTp: computed atlas of surface topography of proteins. *Nucleic Acids Res* 31:3352-3355.

[17] Boeckmann B, Bairoch A, Apweiler R, Blatter M, Estreicher A, Gasteiger E, et al. (2003) The SWISS-PROT protein knowledgebase and its supplement TrEMBL in 2003. *Nucleic Acids Res* 31:365-370.

[18] Bolshakova N, Azuage F. (2003) Cluster validation techniques for genome expression data. *Signal Processing* 83:825-833.

[19] Boutros PC, Okey AB. (2005) Unsupervised pattern recognition: an introduction to the whys and wherefores of clustering microarray data. *Brief Bioinformatics* 6(4):331-343.

[20] Boyle EI, Weng S, Gollub J, Jin H, Botstein D, Cherry JM, Sherlock G. (2004) GO::TermFinder - open source software for accessing Gene Ontology information and finding significantly enriched Gene Ontology terms associated with lists of genes. *Bioinformatics* 20(18):3710-3715.

[21] Bradshaw CR, Surendranath V, Habermann B. (2006) ProFAT: a web-based tool for the functional annotation of protein sequences. *BMC Bioinformatics* 7:466.

[22] Brenner SE, Koehl P, Levitt M. (2000) The ASTRAL compendium for protein structure and sequence analysis. *Nucleic Acids Res* 28:254-256.

[23] Brenner SE. (2001) A tour of structural genomics. *Nat Rev Genetics* 2:801-809.

[24] Brill E. (1995) Transformation-based error-driven learning and natural language processing: a case study in part-of-speech tagging. *Comp Linguistics* 21(4):543-566.

[25] Buchan DW, Shepherd AJ, Lee D, Pearl FM, Rison SC, Thornton JM, Orengo CA. (2002) Gene3D: structural assignment for whole genes and genomes using the CATH domain structure database. *Genome Res* 12:503-514.

[26] Caspi R, Foerster H, Fulcher CA, Kaipa P, Krummenacker M, et al. (2008) The MetaCyc database of metabolic pathways and enzymes and the BioCyc collection of pathway/genome databases. *Nucleic Acids Res* 36:D623-631.

[27] Chandonia J, Brenner SE. (2006) The impact of structural genomics: expectations and outcomes. *Science* 311:347-351.

[28] Chapman BA and Chang JT. (2000) Biopython: Python tools for computational biology. *ACM SIGBIO Newsletter* pp.15-19.

[29] Chen L, Oughtred R, Berman HM, Westbrook J. (2004) TargetDB: a target registration database for structural genomics projects. *Bioinformatics* 20:2860-2862.

[30] Chen H, Sharp BM. (2004) Content-rich biological network constructed by mining PubMed abstracts. *BMC Bioinformatics* 5:147.

[31] Chothia C, Lesk AM. (1986) The relation between the divergence of sequence and structure in proteins. *EMBO J* 5:823-826.

[32] Cimino JJ. (1998) Desiderata for controlled medical vocabularies in the twenty-first century. *Methods Inf Med* 37(4-5): 394-403.

[33] Cohen AM, Hersh WR. (2005) A survey of current work in biomedical text mining. *Brief Bioinform* 6(1):57-71.

[34] Committee IUBMBN. (1992) Enzyme Nomenclature 1992: Recommendations of the Nomenclature Committee of the International Union of Biochemistry and Molecular Biology on the Nomenclature and Classification of Enzymes. San Diego: Academic Press; 1992.

[35] Crespo A, Zhang X, Fernandez A. (2008) Redesigning kinase inhibitors to enhance specificity. *J Med Chem* 51(16):4890-4898.

[36] The DALI Server. Holm L. <http://www.ebi.ac.uk/dali> Accessed April 2007.

[37] Daraselia N, Yuryev A, Egorov S, Mazo I, Ispolatov I. (2007) Automatic extraction of gene ontology annotation and its correlation with clusters in protein networks. *BMC Bioinformatics* 8:243.

[38] de Hoon MJ, Imoto S, Nolan J, and Miyano S. (2004) Open source clustering software. *Bioinformatics* 20(9) 1453-4.

[39] de Hoon M. Cluster 3.0: Open source Clustering software. <http://bonsai.ims.u-tokyo.ac.jp/~mdehoon/software/cluster/software.htm> Accessed September 2008.

[40] Domingos P, Pazzani M. (1997) On the optimality of the simple Bayesian classifier under zero-one loss. *J Mach Learn Res* 29:103-137.

[41] Dougherty WG, Semler BL. (1993) Expression of virus-encoded proteinases: functional and structural similarity to cellular enzymes. *Microbiol Rev* 57(4):781-822.

[42] Druker BJ, Lydon NB. (2000) Lessons learned from the development of an Abl tyrosine kinase inhibitor for chronic myelogenous leukemia. *J Clin Invest* 105(1):3-7.

[43] Ebert JC, Altman RB. (2008) Robust recognition of zinc binding sites in proteins. *Protein Sci* 17(1):54-65.

[44] Eisen MB, Spellman PT, Brown PO, Botstein D. (1998). Cluster analysis and display of genome-wide expression patterns. *Proc Natl Acad Sci* 95:14863-14868.

[45] Fetrow JS, Skolnick J. (1998) Method for prediction of protein function from sequence using the sequence-to-structure-to-function paradigm with application to glutaredoxins/thioredoxins and T1 ribonucleases. *J Mol Biol* 281:949-968.

[46] Fetrow JS, Godzik A, Skolnick J. (1998) Functional analysis of the Escherichia coli genome using the sequence-to-structure-to-function paradigm: identification of proteins exhibiting the glutaredoxin/thioredoxin disulfide oxidoreductase activity. *J Mol Biol* 282:703-711.

[47] Friedberg I, Harder T, Godzik A. (2006) JAFA: a protein function annotation meta-server. *Nucleic Acids Res* 34(Web Server issue):W379-W381.

[48] Friedberg I. (2006) Automated protein function prediction - the genomic challenge. *Brief Bioinformatics* 7(3):225-242.

[49] Frijters R, Heupers B, van Beek P, Bouwhuis M, van Schaik R, de Vlieg J, Polman J, Alkema W. (2008) CoPub: a literature-based keyword enrichment tool for microarray data analysis. *Nucleic Acids Res* 36(Web Server issue):W406-10.

[50] Gabow AP, Leach SM, Baumgartner WA, Hunter LE, Goldberg DS. (2008) Improving protein function prediction methods with integrated literature data. *BMC Bioinformatics* 9:198.

[51] Gelman A, Carlin JB, Stern HS, Rubin DB. Bayesian Data Analysis, 2nd Ed. 2004. *Texts in Statistical Science Series.* Chapman & Hall/CRC, Boca Raton.

[52] Gibbons FD, Roth FP. (2002) Judging the quality of gene expression-based clustering methods using gene annotation. *Genome Res* 12(10):1574-1581.

[53] Glazer DS, Radmer RJ, Altman RB. (2008) Combining molecular dynamics and machine learning to improve protein function recognition. *Pac Symp Biocomput* 332-343.

[54] The GO Consortium. (2000) Gene Ontology: tool for the unification of biology. *Nature Genetics* 25:25-29.

[55] The GO Consortium. (Last modified Sep 26, 2008) Gene Ontology Tools. <http://www.geneontology.org/GO.tools.shtml> Accessed November 17, 2008.

[56] GOA for UniProt, version 60.0. (Release date April 2008) <ftp://ftp.ebi.ac.uk/pub/databases/GO/goa/UNIPROT/gene_association.goa_uniprot.gz> Accessed June 16, 2008.

[57] Greene LH, Lewis TE, Addou S, Cuff A, Dallman T, Dibley M, et al. (2007) The CATH domain structure database: new protocols and classification levels give a more comprehensive resource for exploring evolution. *Nucleic Acids Res* 35:291-297.

[58] Haft DH, Selengut JD, White O. (2003) The TIGRFAMs database of protein families. *Nucleic Acids Res* 31:371-373.

[59] Halperin I, Glazer DS, Wu S, Altman RB. (2008) The FEATURE framework for protein function annotation: modeling new functions, improving performance, and extending to novel applications. *BMC Genomics* 9(Suppl 2):S2.

[60] Handl J, Knowles J, Kell DB. (2005) Computational cluster validation in post-genomic data analysis. *Bioinformatics* 21(15):3201-3212.

[61] Hartigan J. (1975) *Clustering algorithms.* New York: John Wiley & Sons, Inc.

[62] Hendrickson WA. (2007) Impact of structures from the Protein Structure Initiative. *Structure* 15(12):1528-1529.

[63] Henzler RG. (1978) Free or controlled vocabularies: some statistical user-oriented evaluations of biomedical information systems. *Internat Classification* 5(1):21-26.

[64] Hersh WR, Bhupatiraju RT, Ross L, Roberts P, Cohen AM, Kraemer DF. (2006) Enhancing access to the Bibliome: the TREC 2004 Genomics Track. *J Biomed Discovery Collaboration* 1:3.

[65] Hirschman L, Yeh A, Blaschke C, Valencia A. (2005) Overview of BioCreAtivE: critical assessment of information extraction for biology. *BMC Bioinformatics* 6(Suppl 1):S1.

[66] Holm L, Sander C. (1997) Dali/FSSP classification of three-dimensional protein folds. *Nucleic Acids Res* 25:231-234.

[67] Holm RH, Kennepohl P, Solomon EI. (1996) Structural and functional aspects of metal sites in biology. *Chem Rev* 96(7):2239-2314.

[68] Huang DW, Sherman BT, Tan Q, Collins JR, Alvord WG, Roayaei J, Stephens R, Baseler MW, Lane HC, Lempicki RA. (2007) The DAVID gene functional classification tool: a novel biological module-centric algorithm to functionally analyze large gene lists. *Genome Biology* 8:R183.

[69] Hulo N, Bairoch A, Bulliard V, Cerutti L, De Castro E, Langedijk-Genevaux P, Pagni M, Sigrist C. (2006) The PROSITE database. *Nucleic Acids Res* 32:227-230.

[70] Ide NC, Loan RF, Demner-Fushman D. (2007) Essie: a concept-based search engine for structured biomedical text. *J Am Med Inf Assoc* 14(3):253-263.

[71] The InterPro Consortium. (2002) InterPro: an integrated documentation resource for protein families, domains, and functional sites. *Briefings Bioinformatics* 3:225-235.

[72] Ito A. (Last update 2006) Java TreeView. `<http://sourceforge.net/projects/jtreeview/>` Accessed September 2008.

[73] Jaeger S, Gaudan S, Leser U, Rebholz-Schuhmann D. (2008) Integrating protein-protein interactions and text mining for protein function prediction. *BMC Bioinformatics* 9(Suppl 8):S2.

[74] JAligner. Moustafa A. `<http://jaligner.sourceforge.net>` Accessed April 2007.

[75] Jambon M, Imberty A, Deleage G, Geourjon C. (2003) A new bioinformatic approach to detect common 3D sites in protein structures. *Proteins* 52:137-145.

[76] Jensen LJ, Kuhn M, Stark M, Chaffron S, Greevey C, Muller J, et al. (2008) STRING 8 - a global view on proteins and their functional interactions in 630 organisms. *Nucleic Acids Res* 37(Database issue):D412-6.

[77] Jenssen TK, Laegreid A, Komorowski J, Hovig E. (2001) A literature network of human genes for high-throughput analysis of gene expression. *Nature Genetics* 28:21-28.

[78] Jiang J, Zhai CX. (2007) An empirical study of tokenization strategies for biomedical information retrieval. *Inf Retrieval* 10(4-5):341-363.

[79] Juncker AS, Jensen LJ, Pierleoni A, Bernsel A, Tress ML, Bork P, et al. (2009) Sequence-based feature prediction and annotation of proteins. *Genome Biol* 10:206.

[80] Kanehisa M, Goto S, Kawashima S, Okuno Y, Hattori M. (2004) The KEGG resource for deciphering the genome. *Nucleic Acids Res* 32:D277-D280.

[81] Kim SH, Shin DH, Choi IG, Schulze-Gahmen U, Chen S, Kim R. (2003) Structure-based functional inference in structural genomics. *J Struct Funct Genomics* 4(203):129-135.

[82] Kim JD, Ohta T, Tateisi Y, Tsujii J. (2003) GENIA corpus - semantically annotated corpus for bio-text mining. *Bioinformatics* 19(Suppl 1):i180-i182.

[83] Koike A, Niwa Y, Takagi T. (2005) Automatic extraction of gene/protein biological functions from biomedical text. *Bioinformatics* 21(7):1227-1236.

[84] Krallinger M, Valencia A. (2005) Text-mining and information-retrieval services for molecular biology. *Genome Biology* 6:224.

[85] Krissinel E, Henrick K. (2004) Secondary-structure matching (SSM), a new tool for fast protein structure alignment in three dimensions. *Acta Cryst* D12:2256-2268.

[86] Kufareva I, Budagyan L, Raush E, Totrov M, Abagyan R. (2007) PIER: protein interface recognition for structural proteomics. *Proteins* 67(2):400-417.

[87] Landau M, Mayrose I, Rosenberg Y, Glaser F, Martz E, Pupko T, Ben-Tal N. (2005) ConSurf 2005: the projection of evolutionary conservation scores of residues on protein structures. *Nucleic Acids Res* (33 Web Server):W299-302.

[88] Laskowski RA, Chistyakov VV, Thornton JM. (2004) PDBsum more: new summaries and analyses of the known 3D structures of proteins and nucleic acids. *Nucleic Acids Res* 33:D266-D268.

[89] Laskowski RA, Watson JD, Thornton JM. (2005) Protein function prediction using local 3D templates. *J Mol Biol* 351:614-626.

[90] Laskowski RA, Watson JD, Thornton JM. (2005) ProFunc: a server for predicting protein function from structure. *Nucleic Acids Res* 33:89-93.

[91] Lattman E. (2004) The state of the Protein Structure Initiative. *Proteins* 54(4):611-5.

[92] Leser U, Hakenberg J. (2005) What makes a gene name? Named entity recognition in the biomedical literature. *Brief Bioinform* 6(4):357-369.

[93] Levitt M. (2007) Growth of novel protein structural data. *Proc Natl Acad Sci* 104:3183-3188.

[94] Levy R, Edelman M, Sobolev V. (2009) Prediction of 3D metal binding sites from translated gene sequences based on remote-homology templates. *Proteins: Struct Funct Bioinform* Epub ahead of print. DOI: 10.1002/prot.22352.

[95] Liang MP, Brutlag DL, Altman RB. (2003) Automated construction of structural motifs for predicting functional sites on protein structures. *Pac Symp Biocomp* 2003, 204-215.

[96] Liang MP, Banatao DR, Klein TE, Brutlag DL, Altman RB. (2003) WebFEATURE: an interactive web tool for identifying and visualizing functional sites on macromolecular structures. *Nucleic Acids Res* 31(13):3324-3327.

[97] Lin J, Wilbur WJ. (2007) PubMed related articles: a probabilistic topic-based model for content similarity. *BMC Bioinformatics* 8:423.

[98] Lobley AE, Nugent T, Orengo CA, Jones DT. (2008) FFPred: an integrated feature-based function prediction server for vertebrate proteomes. *Nucleic Acids Res* 36(Web server issue):W297-302.

[99] Behnel S, Faassen M, et al. lxml Python library. Version: 2.1.5. `<http://codespeak.net/lxml/>` Accessed March 2009.

[100] MacQueen JB. (1967) Some methods for classification and analysis of multivariate observations. *Proc 5th Berkeley Symp Math Stat Probability* 1:281-297, Univ Calif Press.

[101] Madera M, Vogel C, Kummerfeld SK, Chothia C, Gough J. (2004) The SUPERFAMILY database in 2004: additions and improvements. *Nucleic Acids Res* 32:235-239.

[102] Manning CD, Schütze H. Foundations of Statistical Natural Language Processing. Cambridge: MIT Press. 2001.

[103] Marsden RL, Lewis TA, Orengo CA. (2007) Towards a comprehensive structural coverage of completed genomes: a structural genomics viewpoint. *BMC Bioinformatics* 8:86.

[104] Medvedovic M, Sivaganesan S. (2002) Bayesian infinite mixture model bases clustering of gene expression profiles. *Bioinformatics* 18(9):1194-1206.

[105] National Library of Medicine. (2008) *Medical Subject Headings (MeSH) Fact Sheet.* `<http://www.nlm.nih.gov/pubs/factsheets/mesh.html>` Accessed May 2009.

[106] Messerschmidt A, Huber R. (1990) The blue oxidases, ascorbate oxidase, laccase and ceruloplasmin. *Eur J Biochem* 187:341-352.

[107] Minguez P, Al-Shahrour F, Montaner D, Dopazo J. (2007) Functional profiling of microarray experiments using text-mining derived bioentities. *Bioinformatics* 23(22):3098-3099.

[108] Mooney SD, Liang MP, DeConde R, Altman RB. (2005) Structural characterization of proteins using residue environments. *Proteins* 61:741-747.

[109] Mulder NJ, Apweiler R, Attwood TK, Bairoch A, et al. (2007) New developments in the InterPro database. *Nucleic Acids Res* 35:224-228.

[110] Murzin A, Brenner SE, Hubbard T, Chothia C. (1995) SCOP: a structural classification of proteins database for the investigation of sequences and structures. *J Mol Biol* 247:536-540.

[111] Nair R, Liu J, Soong T, Acton TB, Everett JK, Kouranov A, et al. (2009) Structural genomics is the largest contributor of novel structural leverage. *J Struct Funct Genomics* 10(2):181-191.

[112] Neurath H. (1984) Evolution of proteolytic enzymes. *Science* 224(4647):350-357.

[113] Nimrod G, Schushan M, Steinberg DM, Ben-Tal N. (2008) Detection of functionally important regions in "hypothetical proteins" of known structure. *Structure* 16(12):1755-63.

[114] Okada C, Maegawa Y, Yao M, Tanaka I. (2006) Crystal structure of an RtcB homolog protein (PH1602-extein protein) from *Pyrococcus horikoshii* reveals a novel fold. *Proteins Struct Funct Bioinf* 63(4):1119-1122.

[115] Ouzounis CA, Coulson RMR, Enright AJ, Kunin V, Pereira-Leal JB. (2003) Classification schemes for protein structure and function. *Nat Rev Genetics* 4:508-519.

[116] Pal D, Eisenberg D. (2005) Inference of protein function from protein structure. *Structure* 13:121-130.

[117] PDB statistics: Yearly growth. RCSB Protein Data Bank. `<http://www.pdb.org/pdb/statistics/contentGrowthChart.do?content=total&seqid=100>` Accessed November 2008.

[118] Pegg SC, Brown S, Ojha S, Huang CC, Ferrin TE, Babbitt PC. (2005) Representing structure-function relationships in mechanistically diverse enzyme superfamilies. *Pac Symp Biocomput* 358-369.

[119] Pegg SC, Brown SD, Ojha S, Seffernick J, Meng EC, Morris JH et al. (2006) Leveraging enzyme structure-function relationships for functional inference and experimental design: the Structure-Function Linkage Database. *Biochemistry* 45:2545-2555.

[120] Peters B, Moad C, Youn E, Buffington K, Heiland R, Mooney S. (2006) Identification of similar regions of protein structures using integrated sequence and structure analysis tools. *BMC Struct Biol* 6:4.

[121] Pettit FK, Bare E, Tsai A, Bowie JU. (2007) HotPatch: a statistical approach to finding biologically relevant features on protein surfaces. *J Mol Biol* 369:863-879.

[122] Polacco BJ, Babbitt PC. (2006) Automated discovery of 3D motifs for protein function annotation. *Bioinformatics* 22:723-730.

[123] Porter MF. (1980) An algorithm for suffix stripping. *Program* 14(3):130-137.

[124] Porter CT, Bartlett GJ, Thornton JM. (2004) The Catalytic Site Atlas: a resource of catalytic sites and residues identified in enzymes using structural data. *Nucleic Acids Res* 32:D129-D133.

[125] Prasad TSK, Goel R, Kandasamy K, Keerthikumar S, Telikicherla D, et al. (2008) Human Protein Reference Database - 2009 Update. *Nucleic Acids Res* 37(Database issue):D767-72.

[126] National Library of Medicine and National Institutes of Health. 2008. PubMed. <http://www.ncbi.nlm.nih.gov/pubmed> Accessed November 2008.

[127] Quevillon E, Silventoinen V, Pillai S, Harte N, Mulder NJ, Apweiler R, Lopez R. (2005) InterProScan: protein domains identifier. *Nucleic Acids Res* 33:116-120.

[128] Ramadan NM, Halvorson H, Vande-Linde A, et al. (1989) Low brain magnesium in migraine. *Headache* 29(9):590-593.

[129] Raychaudhuri S, Schutze H, Altman RB. (2002) Using text analysis to identify functionally coherent gene groups. *Genome Res* 12:1582-1590.

[130] Raychaudhuri S, Altman RB. (2003) A literature-based method for assessing the functional coherence of a gene group. *Bioinformatics* 19(3):396-401.

[131] Raychaudhuri S, Chang JT, Imam F, Altman RB. (2003) The computational analysis of scientific literature to define and recognize gene expression clusters. *Nucleic Acids Res* 31(15): 4553-4560.

[132] Reimand J, Kull M, Peterson H, Hansen J, Vilo J. (2007) g:Profiler - a web-based toolset for functional profiling of gene lists from large-scale experiments. *Nucleic Acids Res* 35(Web Server issue):W193-W200.

[133] Rentzsch R, Orengo CA. (2009) Protein function prediction - the power of multiplicity. *Trends Biotech* 27(4):210-219.

[134] Rosenberg JN, Oyler GA, Wilkinson L, Betenbaugh MJ. (2008) A green light for engineered algae: redirecting metabolism to fuel a biotechnology revolution. *Curr Opin Biotech* 19(5):430-436.

[135] Savoy J. (2004) Bibliographic database access using free-text and controlled vocabulary: an evaluation. *Inf Processing Managment* 41(4):873-890.

[136] SeqFEATURE PDB Scan Data <http://feature.stanford.edu/webfeature/data>

[137] Shatkay H, Feldman R. (2003) Mining the biomedical literature in the genomic era: an overview. *J Comp Biol* 10:821-855.

[138] Shaw MP, Bond CS, Roper JR, Gourley DG, Ferguson MA, Hunter WN. (2003) High-resolution crystal structure of Trypanosoma brucei UDP-galactose 4'-empimerase: a potential target for structure-based development of novel trypanocides. *Mol Biochem Parasitol* 126(2):173-80.

[139] Sherlock G. (2000) Analysis of large-scale gene expression data. *Curr Op Immunol* 12(2):201-205.

[140] Sokabe M, Okada A, Yao M, Nakashima T, Tanaka I. (2005) Molecular basis of alanine discrimination in editing site. *Proc Natl Acad Sci* 102(33):11669-11674.

[141] Sonnhammer E, Eddy S, Birney E, Bateman A, Durbin R. (1998) Pfam: multiple sequence alignments and HMM-profiles of protein domains. *Nucleic Acids Res* 26:320-322.

[142] Supplements for Functional Studies Based on High Resolution Structures Obtained in the Protein Structure Initiative. (Released June 25, 2003) National Institute of General Medical Sciences. <http://www.nigms.nih.gov/Initiatives/PSI/Supplements> Accessed November 2008.

[143] Svenonius E. (1986) Unanswered questions in the design of controlled vocabularies. *J Am Soc Inf Science* 37(5):331-340.

[144] Swanson DR. (1986) Fish oil, Raynaud's syndrome, and undiscovered public knowledge. *Perspect Biol Med* 30(1):7-18.

[145] Swanson DR. (1988) Migraine and magnesium: eleven neglected connections. *Perspect Biol Med* 31:526-557.

[146] Tamayo P, Slonim D, Mesirov J, Zhu Q, Kitareewan S, Dmitrovsky E, Lander ES, Golub TR. (1999) Interpreting patterns of gene expression with self-organizing maps: Methods and application to hematopoietic differentiation. *Proc Natl Acad Sci* 96(6):2907-2912.

[147] Tan P, Steinbach M, Kumar V. (2006) *Introduction to Data Mining.* Boston: Addison-Wesley.

[148] TargetDB statistics. TargetDB. <http://targetdb.pdb.org/statistics/pdb_targetdb_title.html> Accessed November 2008.

[149] Thomas PD, Campbell MJ, Kejariwal A, Mi H, Karlak B, Daverman R, et al. (2003) PANTHER: A library of protein families and subfamilies indexed by function. *Genome Res* 13:2129-2141.

[150] 3BJQ structure page. TOPSAN website. <http://www.topsan.org/index.php?title=Proteins/JCSG/3bjq> Accessed November 17, 2008.

[151] The Open Protein Structure Annotation Network (TOPSAN). (2006) <http://www.topsan.org> Accessed November 17, 2008.

[152] Torres-LariosA, Sankaranarayanan R, Rees B, Dock-Bregeon A, Moras D. (2003) Conformational movements and cooperativity upon amino acid, ATP, and tRNA binding in threonyl-tRNA synthetase. *J Mol Biol* 331(1):201-211.

[153] Tseng GC. (2007) Penalized and weighted K-means for clustering with scattered objects and prior information in high-throughput biological data. *Bioinformatics* 23(17):2247-2255.

[154] Uetz P, Giot L, Cagney G, Mansfield TA, Judson RS, Knight JR, Lockshon D, et al. (2000) A comprehensive analysis of protein-protein interactions in Saccharomyces cerevisiae. *Nature* 403:623-627.

[155] The Uniprot Consortium. (2008) The Universal Protein Resource (UniProt) 2009. *Nucleic Acids Res* 37(Database issue):D169-74.

[156] Devos D, Valencia A. (2000) Practical limits of function prediction. *Proteins: Struct Funct Bioinform* 41(1):98-107.

[157] van Aken D, Zevenhoven-Dobbe J, Gorbalenya AE, Snijder EJ. (2006) Proteolytic maturation of replicase polyprotein pp1a by the nsp4 main proteinase is essential for equine arteritis virus replication and includes internal cleavage of nsp7. *J General Virol* 87:3473-3482.

[158] Wallace AC, Borkakoti N, Thornton JM. (1997) TESS: a geometric hashing algorithm for deriving 3D coordinate templates for searching structural databases. Application to enzyme active sites. *Protein Sci* 6:2308-2323.

[159] Watson JD, Sanderson S, Ezersky A, Savchenko A, Edwards A, Orengo C, et al. (2007) Towards fully automated structure-based function prediction in structural genomics: a case study. *J Mol Biol*367:1511-1522.

[160] Weeber M, Klein H, de Jong-van den Berg LTW, Vos R. (2001) Using concepts in literature-based discovery: simulating Swanson's Raynaud-fish oil and migraine-magnesium discoveries. *J Am Soc Inf Sci Tech* 57(7):548-557.

[161] Wei L, Altman RB. (1998) Recognizing protein binding sites using statistical descriptions of their 3D environments. *Pac Symp Biocomp* 497-508.

[162] Wei L, Altman RB. (2003) Recognizing complex, asymmetric functional sites in protein structures using a Bayesian scoring function. *J Bioinform Comput Biol* 1:119-38.

[163] Wilson CA, Kreychman J, Gerstein M. (2000) Assessing annotation transfer for genomics: quantifying the relations between protein sequence, structure and function through traditional and probabilistic scores. *J Mol Biol* 297:233-249.

[164] Wilson RJ, Goodman JL, Strelets VB, and the FlyBase Consortium (2008). FlyBase: integration and improvements to query tools. *Nucleic Acids Res* 36:D588-D593.

[165] Wu S, Liang MP, Altman RB. (2008) The SeqFEATURE library of 3D functional site models: comparison to existing methods and applications to protein function annotation. *Genome Biol* 9(1):R8.

[166] Wu S, Liang MP. WebFEATURE. Updated 2007. `<http://feature.stanford.edu/webfeature>`

[167] Wu S. CYS sub-clusters on WebFEATURE. Created May 2009. `<http://feature.stanford.edu/clustering>`

[168] Xenarios I, Rice DW, Salwinski L, Baron MK, Marcotte EM, Eisenberg D. (2000) DIP: the Database of Interacting Proteins. *Nucleic Acids Res* 28:289-291.

[169] Yoon S, Ebert JC, Chung EY, De Micheli G, Altman RB. (2007) Clustering protein environments for function prediction: finding PROSITE motifs in 3D. *BMC Bioinformatics* 8(Suppl 4):S10.

[170] Youn E, Peters B, Radivojac P, Mooney SD. (2007) Evaluation of features for catalytic residue prediction in novel folds. *Protein Sci* 16:216-226.

[171] Zhang F, Zhan G. (2008) Rational design of an enzyme mutant for anti-cocaine therapeutics. *J Comput Aided Mol Des* 22(9):661-671.