Discovery and Analysis of Aligned Pattern Clusters from Protein Family Sequences

by

En-Shiun Annie Lee

A thesis

presented to the University of Waterloo

in fulfillment of the

thesis requirement for the degree of

Doctor of Philosophy

in

Systems Design Engineering

Waterloo, Ontario, Canada, 2014

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

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

Abstract

Protein sequences are essential for encoding molecular structures and functions. Consequently, biologists invest substantial resources and time discovering functional patterns in proteins. Using high-throughput technologies, biologists are generating an increasing amount of data. Thus, the major challenge in biosequencing today is the ability to conduct data analysis in an efficient and productive manner. Conserved amino acids in proteins reveal important functional domains within protein families. Conversely, less conserved amino acid variations within these protein sequence patterns reveal areas of evolutionary and functional divergence.

Exploring protein families using existing methods such as multiple sequence alignment is computationally expensive, thus pattern search is used. However, at present, combinatorial methods of pattern search generate a large set of solutions, and probabilistic methods require richer representations. They require biological ground truth of the input sequences, such as gene name or taxonomic species, as class labels based on traditional classification practice to train a model for predicting unknown sequences. However, these algorithms are inherently biased by mislabelling and may not be able to reveal class characteristics in a detailed and succinct manner.

A novel pattern representation called an Aligned Pattern Cluster (AP Cluster) as developed in this dissertation is compact yet rich. It captures conservations and variations of amino acids and covers more sequences with lower entropy and greatly reduces the number of patterns. AP Clusters contain statistically significant patterns with variations; their importance has been confirmed by the following biological evidences: 1) Most of

the discovered AP Clusters correspond to binding segments while their aligned columns correspond to binding sites as verified by pFam, PROSITE, and the three-dimensional structure. 2) By compacting strong correlated functional information together, AP Clusters are able to reveal class characteristics for taxonomical classes, gene classes and other functional classes, or incorrect class labelling. 3) Co-occurrence of AP Clusters on the same homologous protein sequences are spatially close in the protein's three-dimensional structure.

These results demonstrate the power and usefulness of AP Clusters. They bring in similar statistically significance patterns with variation together and align them to reveal protein regional functionality, class characteristics, binding and interacting sites for the study of protein-protein and protein-drug interactions, for differentiation of cancer tumour types, targeted gene therapy as well as for drug target discovery.

Acknowledgements

I would like to thank all the important researchers and academics who made this dissertation possible. First I would like to thank my supervisor, Professor Andrew K. C. Wong, and my co-supervisor, Professor Daniel Stashuk, on embarking on this exciting journey of research together. From Professor Wong, I learned the many facets of research and business as well as faith and life. Thank you for your patience, unyielding work ethic, research excellence, and passionate research pursuits. I would like to thank Professor Stashuk for his strongest support in all areas of study and research, including funding and PhD process.

I would also like thank the professors who contributed as collaborators to the collected data in this dissertation: Professor Tun Wen Pai and his team, as well as Professor Dawn Bowdish and her team. Next, I would like thank my external examiner Professor Jimmy Huang and my committee members Professor John Zelek, Professor David Clausi, and Professor Forbes Burkowski. This journey would not have started without my previous masters and undergraduate supervisors: Professor Ming Li, Professor Jeremy Barbay, and Professor Andrew Woolley.

I also thank my colleagues whom have worked tirelessly in publishing research but first most especially Sanderz Fung and Antonio Szeto for their outstanding collaborative work, as well as others (in order of seniority): Dr. Gary Li, Dennis Zhang, Pei-Yuan Chou, Fiona Whelan, Franky Yeung, Ben Yu; and also the bright and eager research assistants who are full of potential: Elizabeth Yam, Eelizabeth Chan, Fatemeh Jahed, Maryam Jahed, Jannah Henzl, William Fu, James Han, Prashant Bharadwaj, Helen Lu, and Eva Ho. Working with current and past members of the centre of Pattern Analysis and Machine Intelligence has

been a pleasure, such as Professor Dana Kulic, Dr. Yang Wang (PDT), Professor Adams Kong, Yun-Qian (Mike) Miao, Sepideh Seifzadeh, Meena Abdel-Maseeh, Jonathan Lin, Rosalind Klein, and many others. As well as colleagues in the Visual Image Process Lab Professor Alexander Wong, Fan Li, and Lei Wang. I must also thank those who advised and mentored me here at Waterloo: Professor Keith Hipel, Professor Sze Sze Chan, and Dr. Steve Wong; as well as my female mentors: Professor Marie DesJardins, Professor Anne Condon, and Professor Christina Boucher. I must thank my colleagues for critical feedback through the publication process, Dr. Kirk Dirstin, Denis Yeun, and many working and graduate colleagues from Awesome cell, mCCF, and graduate cell.

Lastly, I would like to thank the professors in the Systems Design Engineering department whom I had been a teaching assistant or gave me academic advice; as well as the graduate students and academic staff. Thank you for making this experience enriching and fruitful.

Dedication

I dedicated this work to my earthly family. To the Lee family and Chan family, as well as surrogate Wai family. For your cares here on earth, I know I am well loved. Lee Family: (Hong-Ter Lee, Shiow-Chin Lee, Charles Lee, Chun May Li) Chan Family: (Mom and Dad, Ada Chan (and family), Eva Chan, Daniel Chan) Wai Family: (Tim Wai, Christina Waters, Davina Leong, Emily Wong, Clare Mok, Oreo Wai) Also for the support from Awesome Cell, Graduate cellgroup (the oldies), and mCCF. "May (you) be a light ... in dark places, when all other lights go out." – J.R.R. Tolkien, The Two Towers Awesome cell: Oliver Wong, Kevin Chan, Vivian Chan, Jonathan Lin, Stephanie Lu, Gregory Lui, Didi Cheung, Josh Lo, Alan Mak, Ivy Lam, Jorge Quan, Mark Tse, Clare Mok, and countless others; Graduate cell: Heidi and Waiki Lee, Emily and Joseph Wong, Alex and Lianna Chinn, Maimie Tou, Kenneth Ho. mCCF: Winston Hsieh, Rebecca Lee, Francis Cheng, Betty Chang (David Lee), Hans Lin, William Fu, Andy Huang, Alice Lee, and Jennifer Kuo.

This work is dedicated to the almighty creator without whom nothing is possible, to Him be the kingdom and the glory. For those who came before; for those here now; for future generations to come. And just as it had started with this journey, "In the beginning was the Word" John 1:1.

Contents

List of Tables

List of Figures

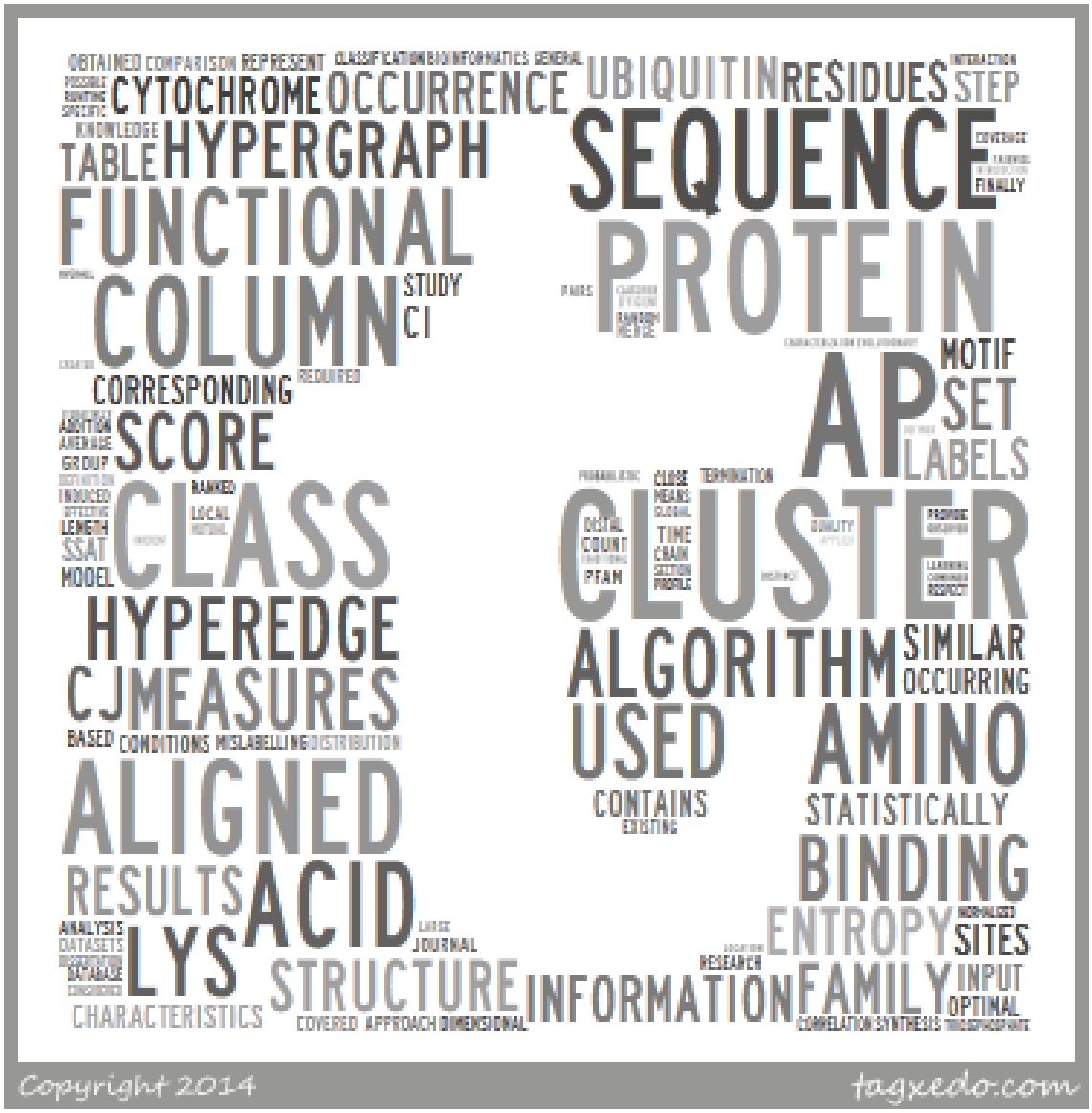


Figure 1: An word graphic of the common words used in this dissertation.

Chapter 1

Introduction

> I have called this principle, by which each slight variation, if useful, is preserved, by the term of Natural Selection.
>
> Charles Darwin

Proteins are crucial in the biological functions for all living organisms, including human. Protein sequences are essential in encoding their molecular structures and functions. Consequently, biologists invest substantial resources and time discovering functional patterns in proteins. The human genome was encoded in a decade-long race between scientists and industry in the 1990s, and today, the ENCODE (Encyclopedia Of DNA Elements) project, attempts to identify all functional elements in the human genome sequence. We are now able to sequence genomes quickly and economically; the next generation sequencing is now posed at the edge of scientific discoveries. Using high-throughput technologies, biologists

are generating an increasing amount of data. Thus, the major challenge in acquiring useful results in biosequencing today is the ability to conduct data analysis in order to discover useful knowledge from them in an efficient and productive way.

As the amount of the biomolecular data is becoming larger and our understanding and the use of the inherent patterns in the data become more complex and diverse, new methods are needed to acquire knowledge from the data quickly and effectively. Today, exploring useful knowledge from protein families using existing methods, such as multiple sequence alignment, is computationally expensive; thus, pattern search is used. However, at present, combinatorial methods of pattern search generate a large set of solutions, and probabilistic methods require richer representations. In this approach, biological ground truth of the input sequences, such as gene name or taxonomic species, can be used as class labels in traditional classification are required to train a model for predicting novel sequences. However, these algorithms are inherently biased by the training set. Supervised methods of classifying proteins relying on class labels are often affected by mislabelling of the class labels in the training set or by unbalanced classes. Among the unsupervised approaches, the clustering methods of sequence analysis focus on discovering conserved regions of importance. In global alignment, multiple sequence alignment is time consuming and inaccurate for divergent sequences. On the other hand, motif finding does not represent the patterns adequately for discovering further knowledge. In most of the clustering methods, they try to cluster the protein sequences into clusters which might be related to protein classes. However, as we have observed, different regions may have somewhat different characteristics when relating to protein classes. It is very difficult to know a priori how different functional regions are associated with class variation. We have to rely on the

data to reveal these subtle relations.

The objective of this dissertation research is to acquire useful information from large multiple protein sequence datasets. Aligning and clustering patterns attempt to extract knowledge from data alone and is a paradigm shift from current pattern analysis approaches. An advantage of this approach is that it does not rely on *a priori* knowledge, wet-lab experiments, or stringent constraints to acquire useful information. In fact, it is a data-driven method with strong statistical backing and algorithmic efficacy to render knowledge-rich representation quickly, accurately and comprehensively. It is a paradigm shift from the current pattern analysis approaches. This dissertation explores and follows the new data to knowledge paradigm in proteomic application; that is, it attempts to acquire protein knowledge directly from protein sequence data. It is intended to create a rich yet compact pattern representation capturing conservations and variations in a protein sequence family to reveal biological functionality as well as class characteristics at the amino acid association level within different protein sequence regions without relying on class knowledge or being biased by unknown governing class characteristics inherent in the data. After patterns in the data are discovered, their significance and connection to the biological data can then be interpreted with the support of statistical and functional homology. In this dissertation, three major research questions are addressed and answered:

1. Are the discovered AP Clusters, patterns with variations, biologically meaningful and important? Do the discovered AP Clusters correspond to binding segments and their aligned columns correspond to binding sites? The answer obtained from the research question is affirmative as both claims are confirmed by pFam, PROSITE,

and proteins' three-dimensional structure.

2. Are the variations significantly found in AP Clusters (in patterns, sites, and amino acids within certain sites) reflect biologically important class characteristics? The answer is that biological ground truth of the input sequences, such as gene names or taxonomic species, when used as class labels, does verify the above claims in this dissertation through the use of cluster validity measures which are unaffected by data biases.

3. Are the highly co-occurring AP Clusters discovered within a homologous protein structurally and functionally significant? The answer from this dissertation is confirmed through the discovery and clustering of distant AP Clusters on the same homologous protein sequence based on their co-occurrences. They are by and large, spatially close in the proteins' three-dimensional structure and/or contain molecular interacting functionality as reported in biology literatures.

Sequence analysis is at the heart of our understanding of protein functions since biological data need to be organized to infer biological knowledge. This data-driven method can discover specific target motifs, amino acids characteristics, and motif associations to benefit the scientific community as well as the healthcare and drug industry. The results demonstrate the power of AP Clusters in revealing protein-drug interactions for drug research and to differentiate cancer tumors for targeted gene therapy.

1.1 Thesis Contributions

The contributions of this dissertation can be broken down into three sections. Section I, covered by Chapter 3 and Chapter 4, addresses the most fundamental notion of the dissertation, Aligned Pattern Clusters. It is on the alignment and clustering of statistically significant and non-redundant sequence patterns which are obtained from a previously developed algorithm. It also encompasses its dual representation in the induced data space as well as in a compact structural representation known as an Aligned Pattern Directed Hypergraph. Section II, covered in Chapter 5, brings out the important contribution of Aligned Pattern Directed Hypergraphs in revealing the class characteristics of different protein regions without relying on class labels and prior knowledge. It is unsupervised, data driven, unbiased by the input data, such as mislabelling, unbalanced classes and insufficient or incorrect class information. Section III, covered by Chapter 6, shows another novel contribution in revealing joint functionality of different regions of proteins through clustering via co-occurrence score the discovered Aligned Pattern Clusters. The results of revealing region to regions interaction and/or bindings have significant impact in the study of macromolecular interaction and drug discovery.

1.1.1 Section I of the Dissertation: Pattern Clustering and Representations

A binding site is a region in in protein that chemically binds another molecule called the ligand. Binding sites are typically the functional focus of a protein, and therefore, recog-

nizing them is essential in protein function analysis. Although each protein of the same protein family performs the same function, there are variations amongst the amino acids across each primary sequence. Hence, the conserved amino acid associations across the protein sequences from one protein family reflect its important functions. Similarly, the ubiquitin protein, which forms an ubiquitin chain to regulate processes, contains seven binding residues that are also surrounded by binding segments. These binding residues and segments function by linking individual ubiquitins to create a unique poly-ubiquitin that can be recognized by other ubiquitins. Linking of these binding proteins is directly involved in the control of cancer progression [82]. A common approach to study a protein family's function is to find sequence patterns that have variations. Functional patterns can mutate through evolution [37, 30]; thus each occurrence of the pattern may not be an exact replica at the same location. Hence it is difficult to find and locate the segments that embed the binding residues. In bioinformatics, the two common approaches for identifying a protein family's functions are by multiple sequence alignment and by motif finding. A motif is a sequence pattern that has biological significance, and is thus typically statistically significant. Multiple sequence alignment aligns a set of protein sequences from the same protein family in order to identify important regions and sites in the resulting alignment. Common multiple sequence alignments include Clustal Omega[186], T-Coffee[139], DIALIGN[8] and HMMER[60]. However, finding the global optimal alignment is computationally expensive, and is known in computational complexity analysis as an NP-complete problem [202]. Even with approximate heuristics added, multiple sequence alignment is not efficient in handling large datasets. Moreover, this approach is only appropriate for highly similar sequences, and not for sequences with considerable dissimilarity. Therefore,

instead of aligning the entire sequence globally, it is only suitable to identify similarities locally. Thus, the suspected consensus regions have to be first located and preprocessed ahead of alignment. Another approach for identifying a protein family's function by similar local subsequences [69] is called motif finding, which builds motifs into combinatorial models or probabilistic models. The combinatorial model identifies commonly repeated sequence patterns exhaustively [31, 81, 142]. Work reported in Pevzner et al. [148] and Mandoiu et al. [131] created cliques where vertices are sequence patterns, edges connect similar sequence patterns, and complete graphs represent the best consensus patterns. However, these combinatorial methods are computationally intensive [120] and produce too many likely candidates. The probabilistic model commonly uses the position weight matrix (PWM), which estimates an amino acid distribution at each position while assuming that each position is independent [3, 87]. An alternative random sequence synthesis takes further frame-shifted position into consideration by optimally aligning amino acids to create a probabilistic sequence [39, 213]. Other probabilistic methods make use of the Markov model, where the current state depends on a specified set of past states. One such example is the popular pFam database [175], which builds a profile Hidden Markov Model (HMM) from the multiple sequence alignment of a protein family for classifying proteins and predicting their functionality.

1.1.2 Section II of the Dissertation: Class Characteristics

Two well adopted algorithms of protein sequence classification are the Hidden Markov Models (HMMs) and the Support Vector Machines (SVMs). They have been used to clas-

sify protein sequences based on their class labels. Both use training sequences with known class labels (e.g., protein family, gene function or taxonomic species) to train a classification model and use it later to predict the class of a new protein sequence. However, training a model may pose a methodological problem because biological class labels may be difficult or impossible to acquire and, sometimes, may be changed base on new understandings [209]. As a consequence, the classification accuracy of these methods is often affected by incorrect or insufficient class information, such as mislabelling, incorrect partitioning, and imbalanced class samples.

Because of these recognized problems, some researchers prefer to use unsupervised clustering algorithms to identify amino acid variations without being affected by training error [141]. Traditional methods obtain clusters based on the entire protein sequences encounter new class association problems, since functional complexity of proteins causes functional regions to have multiple functional class characteristics. Hence, a more appropriate approach to study how protein functionality is related to its class characteristics is to first identify homologous local functional regions before exploring how the regional functionality related to its inherent class.

Consequently, we summarize the main contribution of this dissertation section in revealing class characteristics as follows:

1. The use of Aligned Pattern Clusters (AP Clusters) reveals class characteristics [115]: The statistical significant patterns computed in linear time and space when aligned and clustered into AP Clusters could reveal the functional and class characteristics of critical protein regions without relying on prior knowledge by taking advantage of

their patterns and the aligned amino acid variations.

2. External cluster validity measures: Since AP Clusters aggregated the most important functional information into critical regions, incorporating external class labels related distinct AP Cluster representations (patterns, columns, and amino acid variations) to different class distributions. Among these three external measures, the class entropy of a particular representation reveals how different representations of AP Clusters are related to class labels. The next two measures, namely Class Information Gain and Class Mutual Information, related the column of aligned amino acids to its class labels.

3. Internal cluster validity measures: By bringing and compacting relevant functional information into AP Clusters, the three internal measures, (1) Entropy Redundancy, (2) Normalized Sum of Mutual Information Redundancy, and (3) Normalized Sum of Information Gain, are computed from the data to reveal the columns with class distribution so as to identify those that may reveal the underlying patterns or residues corresponding to a single class or various classes. Such an an approach is unsupervised and data-driven.

1.1.3 Section III of the Dissertation: Co-Occurrence of AP Clusters

No specific methods are available to indicate which amino acids in the pattern are not statistically or functionally significant in such models. Consequently, the Aligned Pattern

Cluster (AP Cluster) was introduced in our previous work [116] to provide a knowledge-rich representation of functional regions, by capturing their statistically significant associations of the residues along the sequences and the distribution of their occurrence on each of their aligned segment regions.

With this novel representation, we are now able to study and exploit the co-occurrence to identify binding sites within a protein, between two interacting proteins [123, 93], and between protein and DNA [119, 41]. Here, we define co-occurring patterns as patterns occurring on the same protein sequence. Related works [204, 35, 135] suggests that co-occurring (correlated) residues can provide insights into protein structures. Their hypothesis is that if two residues of a protein form a contact, an amino acid substitution at one position is expected to be compensated by a substitution in another position over the evolutionary time-scale. However, there are far too many co-occurring patterns or residues to consider since the number of patterns discovered and residues as well as their correlations are enormous. Hence, the major drawback of these approaches is that a large number (i.e., the order of 1,000) of homologous and non-redundant protein sequences are required to learn the underlying statistical model [204, 35].

Also, regarding studies on protein families using Evolutionary Tracing (ET) [124], the presence or absence of certain clusters of residue on a protein sequence is a main cause of divergence between globally-specific functions and family-specific functions [129]. Mutagenesis data is required for their studies, and their results suggest that the presence or absence of co-occurring patterns is likely to be linked to functional divergence [129].

In this aspect, a third important contribution of this dissertation is to provide efficient

solutions in answering the following two questions: How can we efficiently discover the frequently co-occurring patterns, given only multiple homologous proteins sequences as input? And what are the biological reasons for their high co-occurrence and how can we relate the pattern co-occurrence findings to the underlying reasons? Our hypothesis is that co-occurring patterns can reflect joint functionality. They might have formed chemical bonds, or they need to co-operate on certain biological functions. We started our study by collecting homologous protein sequences from protein databases. We developed an efficient algorithm based on our previous work [218, 115] to identify the frequently co-occurring patterns using only sequence data as input. We verified our results by computing spatial distances between co-occurring patterns using the corresponding 3D structures. We also surveyed the literature to find additional biological evidence to support the notion of co-occurrence.

In view of the above observation and experimental results obtained, the contribution of this dissertation is three-fold. First, we established a framework to study functional regions of proteins by exploiting the co-occurrences of patterns to reveal concurrent distant functions and structural relations. To our knowledge, this is the first study to identify co-occurrence of patterns rather than amino acids using only homologous protein sequences as input. Second, we developed an algorithm that is statistically reliable, efficient, and visualizable (in domain location, structural and functional relation, amino acid conservation and variations) as an integrated process. Compared to existing algorithms studying correlations (in residues), our algorithm is novel in that it does not require a large amount of homologous protein sequences to identify co-occurrences (of patterns) through training. Third, our discovered co-occurrences of patterns that are novel to the biological community

will provide new insights to their studies of biological functions.

1.2 Thesis Organization

The overall structure of the dissertation includes an introduction, a background, and the remaining chapters divided into three sections: 1) Clustering and Representing Patterns (Chapters 3, 4); 2) Class Characterization (Chapter 5); and 3) Co-occurring Patterns (Chapter 6). The contents of these chapters are summarized below with their inter-dependencies (Fig. 1.1).

Section I of our Aligned Pattern Synthesis Process, as illustrated by the text example, synthesizes the data by the following steps: the Pattern Discovery Step, the AP Clustering Step, the Graph Construction Step,and the AP Cluster Refinement Step.

During the Pattern Discovery Step, we discover amongst the family of sequences the most important sequence patterns, which are non-redundant statistically significant associations of amino acids. In the AP Clustering Step, we group and align these patterns into clusters even though the occurrences of the pattern might start at different positions in their input sequences. This step synthesizes patterns with variations without having to search the original input data exhaustively. From this cluster, we synthesize the results into a probabilistic structural pattern known as an AP Hypergraph; this is accomplished in the Graph Construction Step. In the AP Cluster Refinement Step, we extend the AP Clusters into Weak AP Clusters to increase the coverage and then into Conserved AP Clusters to increase the coverage while maintaining Shannon's information entropy.

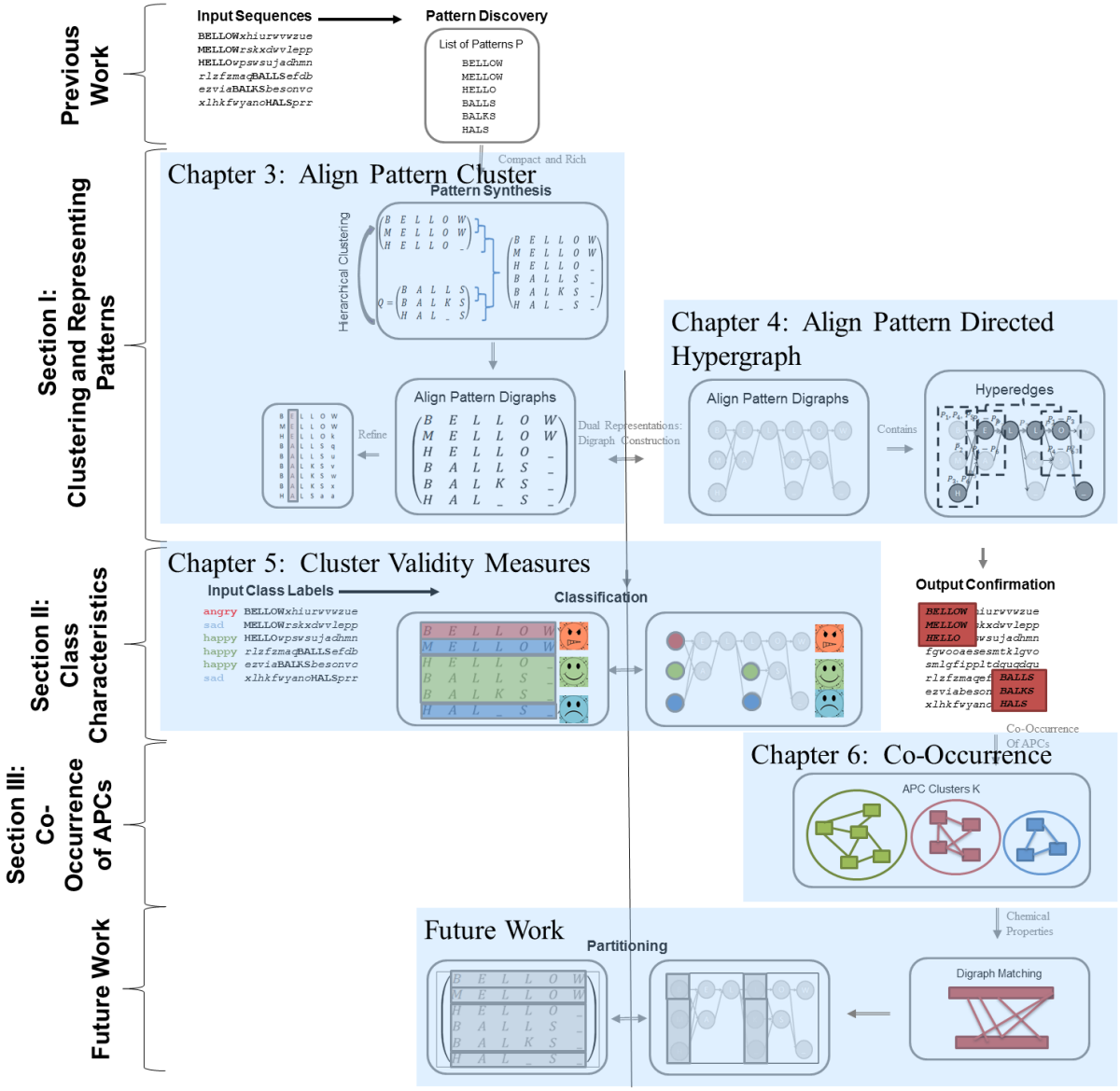


Figure 1.1: Overview of the AP Synthesis Process applied to the text example. The text example using the English alphabet will be used repeatedly throughout the dissertation.

In addition, two steps inferring knowledge from the data Class Characterization Step and Co-Occurrence Step. Next, in the Class Characterization Step of Section II, we verify the AP Clusters by six cluster validity measures: three external and three internal ones. Finally, in the Co-Occurrence Step of Section III, we exploit co-occurring AP Clusters on the same protein sequence to identify functional regions. We develop an efficient algorithm to identify the frequently co-occurring patterns using only homologous protein sequences as input.

1.3 List of Publications

Journal publications related to this dissertation:

1. Pre-Thesis: Andrew K.C. Wong, Dennis Zhuang, Gary C.L. Li, and En-Shiun Annie Lee. "Discovery of Delta Closed Patterns and Non-induced Patterns from Sequences", IEEE Transactions on Knowledge and Data Engineering, 24(8), pp. 1408-1421, 2012.

2. Chapter 3: Andrew K.C. Wong, and En-Shiun Annie Lee. "Aligning and Clustering Patterns to Reveal the Functionality of Biosequences", IEEE Transactions on Computational Biology and Bioinformatics, 2013 (accepted to appear).

3. Chapter 4: En-Shiun Annie Lee, and Andrew KC Wong. "Ranking and compacting binding segments of protein families using aligned pattern clusters." Proteome Science 11.Suppl 1 (2013): S8.

4. Chapter 5: En-Shiun Annie Lee, and Andrew K.C. Wong. "Revealing and Validating Protein Class Characteristics from Align Pattern Clusters", BMC Bioinformatics, 2014 (in progress).

5. Chapter 6: En-Shiun Annie Lee, Sanderz Fung, Ho-Yin Sze-To, and Andrew K. C. Wong. "Discovering co-occurring patterns and their biological significance in protein families", BMC Bioinformatics, 2014 (Selected for Special Journal edition from BIBM)

Chapter 2

Literature Review

Biological sequences control many elements of life, from gene expression to enzymatic reactions. In bioinformatics, revealing the functionality of a protein family's biological significance often requires examining its primary sequences. Identifying conserved sequence patterns in proteins is important for the study of essential protein functions. Furthermore, mutated amino acids in these conserved regions may reflect special functionality that has evolutionarily diverged into sub-families [51]. The structure of this literature review is as follows: (1) experiments and databases, (2) pattern representation and visualization, (3) multiple sequence alignment, (4) motif finding (discover patterns with variations), (5) supervised classification, and (6) co-occurring patterns. They are not partitioned by Section I, II, III because these topics are procedurally progressive and may relate to more than one section.

1. Experiments and Databases: Traditional experiments are time and labor intensive;

therefore, their results are stored into databases. These databases store information either from human-curated experimental results or from annotation algorithms. Even though databases are useful search tools, they cannot discover new knowledge; therefore, sequence analysis techniques like Aligned Pattern Cluster (AP Cluster) as proposed in this dissertation are employed.

2. Multiple sequence alignment: Two traditional sequence analysis approaches are multiple sequence alignment and motif finding. Multiple sequence alignment employs a full set of protein sequences for the entire sequence. However, the alignment depends on parameter optimization and requires input sequences to be non-divergent with high similarity.

3. Motif finding: To find local similarity, motif finding discovers patterns either sequentially, probabilistically, or graphically. Motif finding assumes fixed length, number of mutations, and distance of mutations. Probabilistic patterns compress the representations into over-simplified random variables. Lastly, graphical patterns are computationally intensive. To overcome these shortcomings, AP Cluster is proposed, which furnishes a flexible, rich, yet fast algorithm for representing patterns with variations.

4. Classification: Protein sequence classification is by and large based on supervised methods, such as Hidden Markov Models (HMMs) and Support Vector Machines (SVMs), which use a training dataset to build a model for predicting the testing dataset. Any errors in the training dataset, such as mislabelling, incorrect partitioning, or unbalanced datasets, will affect the accuracy of the supervised algorithms.

This can be demonstrated in a synthetic dataset with errors injected. Therefore, unsupervised clustering with class validity measures for AP Cluster is proposed in this dissertation instead. Further experiments found a trade-off between predictive accuracy and algorithmic speed, both of which AP Cluster overcomes.

5. Co-occurring patterns: Existing methods such as Evolutionary Tracing requires pre-aligned sequences and mutagenesis requires a large amount of input sequences, both of which are time and data intensive. Alternatively, AP Cluster only considers the functional regions and relates them by their induced data, which require minimal dataset due to the fast regional pattern alignment.

Although there are several excellent general reviews of pattern discovery, clustering, and classification in data mining, machine learning, and bioinformatics, to some extent, each reflects a particular application domain and research perspective. Due to the pace of development and breadth of proteomic research, a truly comprehensive review is beyond the scope of this dissertation. Only a brief review is given in this chapter.

2.1 Wetlab Experiments and Protein Databases

The underlying belief of evolution is that amino acids in functional regions are under selection pressure to maintain their functional integrity and thus undergo fewer mutations than less functionally important ones [124]. While wetlab experiments, such as alanine scan, x-ray crystallography, and mass spectroscopy, are labor and time intensive, experimental annotations are saved into databases for future searches. Although database scanning can

be used for finding protein annotations, no novel knowledge discovery can be made. In this dissertation, the protein sequence datasets were extracted from databases including the Pfam and UniProt [17]. Pfam [63] sequences are built from multiple sequence alignments with the help of hidden Markov model; thus, the sequences have been pre-processed for correctness. UniProt sequences are collected from a string query search of the database, so the quality of the sequences depends on the search terms. Therefore, the sequence quality of UniProt is less consistent than Pfam. Additional a priori information for confirming the computational results are taken from PROSITE, SCOP, CATH, and Protein Databank (PDB) [20]. These protein databases are described in greater detail below.

The Protein Databank (PDB) [20] contains the coordinate files of three-dimensional structure of all possible proteins that have been crystallized by experiments. The methodology for obtaining these structures includes nuclear magnetic resonance (NMR), electron microscopy, and x-ray diffraction. UniProt [17] database stores the primary sequences of these protein structures with functional information. The protein information annotated in this database includes protein function, enzyme information, protein-protein interaction, patterns, domains, and binding sites. In addition, UniProt Taxonomy allows the identification of each species' taxonomy, which acts as the biological class label for the protein sequence. Hence, these protein databases provide biological background information: the PDB provides three-dimensional structural information and UniProt provides the biologically annotated information, such as binding site and taxonomy.

Proteins are grouped by evolutionary relatedness, i.e., common ancestry of descent, into protein families that have similar sequences and functions. Several databases describe protein families: PROSITE, SCOP, CATH, and Pfam. First, PROSITE [16] is a database of

known homologous protein motifs compiled by biologists who have annotated the database with motifs of biological significance, such as active sites or binding sites. However, expert curation of biological knowledge limits the expansion of the database. The PROSITE text search allows complex regular expressions for its input. Next, Structural Classification of Proteins (SCOP) [9] is a protein family database that is primarily manually currated into structural classes based on structure and sequence similarities. Then, Class Architecture Topology Homologous Super-family (CATH) [45] is a protein family database based on semi-automatic classification to group the proteins. Finally, Pfam [63] is a protein family database that classifies the proteins by multiple sequence alignments using hidden Markov models.

2.2 Multiple Sequence Alignment

Sequence analysis is based on the assumption that evolutionarily conserved sequences are more similar. Multiple sequence alignment (MSA) takes a set of sequences and lines up the same amino acids in the same vertical columns by adding wildcard characters, which matches any single character in the alphabet, and gaps, which is a null character, into the sequences. Types of MSA algorithms range from exact alignment to heuristic algorithms, which are trade-offs between runtime and accuracy. Exact alignments are slow (i.e., $O(n^k)$ for k sequences n long) but accurate. Therefore, progressive alignments, such as ClustalW [108], use an adjusted score for faster approximate solution; however, initialization errors are propagated. It is not mentioned if ClustalW can benefit from randomized starting sequences. To overcome propagated errors, consistency alignment, such as T-Coffee [139]

uses a library combining local and global alignment that runs $O(n)$ times slower. Probabilistic alignment, such as ProbCons [55], is known to have an increase in sensitivity and accuracy. AP Clusters discover significant patterns, and then aligns and clusters them into similarity groups; thus, combinatorial complexity in matching is drastically reduced.

Based on the algorithm and the set of sequences inputs, there is no correct alignment but only optimal alignments according to the score and termination. Thus, parameters that affect the optimal alignment include: (1) input sequences, (2) score (similarity and penalty), (3) objective function, (4) algorithm, and (5) termination conditions. To begin, the set of sequences that is selected for MSA will affect the quality of the results. In general, sequences with low similarity cannot be aligned, and thus, some practical strategies such as sub-grouping will pre-align sequences. Secondly, the score's similarity and penalty directly affect the optimization of the algorithm. For the similarity score, a matrix of similarity among amino acids may be used, but it depends on the evolutionary similarity defined by the user. A penalty is subtracted from the score when gaps are introduced into the alignment because gaps increase uncertainty in the alignment by adding flexibility to the alignment. Fixed, affine, position-specific, and residue-specific penalties do not add any additional runtime to the algorithm; however, linear affine penalties add additional $O(n)$ to the runtime. Thirdly, the sequence alignment algorithm can be measured by several possible objective functions: sum-of-pairs, relative entropy, matrix distance, and normalized matrix distance. Also, additional information may be used to optimize the algorithm, such as structural information, reading frame, and phylogenetic descent. Finally, as discovered in this dissertation, the type of the algorithm and how to terminate the algorithm both affect the resulting alignment as well. These two additional parameters are

presented in the next chapter (Section I).

Our AP Cluster considers the optimality of the result by comparing entropy of the results by (1) similarity score, (2) alignment algorithm, and (3) termination condition. MSAs are evaluated by a set of benchmarks called BaliBASE [188] that contain a set of true alignments with known three-dimensional structure and conserved regions of the sequence. The evaluation concludes that MSA is not effective for aligning highly diverged sequences that may share only limited regions of conservation. For example, sequences may be derived from ancient recombination events where only a single functional domain is shared. Therefore, local conservations, such as motif finding, is employed instead.

2.3 Pattern Discovery

Unlike MSA that aligns the full sequence, pattern discovery is able to analyze highly diverged sequences and find the limited regions of conservation through the similar patterns it discovers. In motif finding, the input is a group of sequences and a pattern, called the motif, identified in each sequence at a non-fixed position. Motifs can be rigid without any degeneracies or variations in the sequence or flexible, allowing some positions to be wildcards or distant matches. The brute force method of solving flexible motifs is consider NP-hard [121]. Thus, the better approach is to generate motifs and score them based on the amount of over-representation (i.e., surprise). In this manner, the highest scoring motif is the most meaningful. Several motif finding methods exist: iterative, profile, and graphical. Iterative methods can be (1) bottom-up such as consensus, which greedily adds motifs, or BLOCKS [77], which generate ungapped blocks; or (2) top-down such as MEME

[14], which finds the expected motif to maximize the composition iterative, or Gibbs [109], which use randomly generated solutions are improved iteratively until a local optimum. Random projection [32] uses MEME and Gibbs starting points and randomly selected fixed positions. In addition qPMS7 [54] varies q positions of an n-length sequence by no more than distance d. Both of these heuristic methods are fast, but result in low-quality motifs. Finally, YMF [173] discovers significant patterns without redundancy and SP-STAR [32] has a specialized score for subtle signals. Probabilistic methods include Pfam's HMM [175] and random graph [213, 39], which represent the pattern as a sequential graph. Finally the graphical method, WINNOWER [149], builds cliques and finds a consensus. While sequential methods are bounded by fixed parameters, probabilistic methods are over-simplified, and graphical methods are slow. The review from Tompa [190] comparing the motif finding algorithms ranking them from worst to best as follows: CONSENSUS, MEME, YMF, and Weeder [143]; and the recently published qPMS7 claims to be five times faster.

AP Cluster (Chapter 3) discovers statistically significant patterns and reduces results by statistical pruning. It is fast due to its linear time and space algorithm, which is confirmed by runtime comparison with other methods. The algorithm allows flexible length and variations, and allows sensitivity trade-off between coverage and entropy.

2.4 Class Characterization

Machine learning can be divided into two categories of algorithms, supervised learning and unsupervised learning algorithms based on the algorithm's dependence on external class

labels. Supervised learning trains a model with desirable class labels to predict future test samples. Unsupervised learning on the other hand, examines the data itself without class labels, and creates a model from the given data alone. Its effectiveness of grouping data according to the group characteristics are usually evaluated through cases with known class labels.

Supervised learning can be thought of as classification. Typically, the data is separated into training and testing sets. Each sample given with X as the data and Y as data used for prediction. Protein sequence classification trains a model using training sequences with known class labels (e.g., gene function or taxonomic species) to predict the class labels of new protein sequences. Traditional classification algorithms include decision trees, which can be extended into random forests, neural networks, and Bayesian methods. However, these algorithms use class labels exclusively to train the model. However, training a model is challenging because class labels may be difficult or impossible to acquire and, sometimes, may not even be correct. The accuracy of these supervised learning algorithms is significantly affected by incorrect and changing class labels that occur much more often in omics research.

Two existing sequential algorithms classify protein sequences based on their class labels: HMM and the Support Vector Machine (SVM). HMMs are slow and accurate, whereas SVMs are fast, but their accuracy depend on the kernel, which is a sequence similarity score; AP Cluster is much faster than both. When compared to AP Cluster, the accuracy of these methods is effected by mislabelling and unbalance classes, whereas AP Cluster describe the cluster with respect to these biases.

To address this question, unsupervised learning employing clusters with validity measures avoid input class label biases. While previous work uses class labels against the data in normalized point mutual information [201], AP Cluster utilize Class Information Gain and Normalized Sum of Mutual Information Redundancy. Information gain has been used in HMM for selecting a statistically significant model [38], for comparing information gain between variables [73], and as Markov blankets for removing redundant variables without loosing information [98]. Existing work in cluster validity measures for continuous variables compare their maximal information-based non-parameteric exploration (MEME) and MIC (maximal information coefficient) against different measures, Peasron, Spearman, Mutual Information, Core GC, and Correlation, where the functional relationship provides scores that roughly equal the coefficient of determination (R2) and is applicable in various biological datasets [156].

Finally, part of unsupervised learning is the problem of dimensionality reduction, where a large problem with many dimensions is reduced into a smaller problem with less dimensions. We are furthering developing principal component analysis with spectral analysis for identifying chemical properties of important sites [165]. As dimensionality increases, complexity and diversity also increases, and different functional regions may have different functional characteristics related to different kinds of protein classes. Clustering proteins based on the entire sequences may mix different functional groups. For instance, certain functional characteristics in certain regions may be more likely related to gene classes, taxonomical classes, or others. Therefore, AP Cluster is introduced to first identify a homologous local functional regions before relating protein functionality to its class characteristics.

2.5 Co-Occurrence

The final chapter of this dissertation exploits co-occurrence to identify binding sites within a protein, between two interacting proteins [123, 93], and between a protein and a DNA [119, 41]. Here, we define co-occurring patterns as patterns occurring on the same protein sequence. Related works [204, 35, 135] suggests that co-occurring (correlated) residues can provide insights on protein structures. Their hypothesis is that if two residues of a protein form a contact, an amino acid substitution at one position is expected to be compensated by a substitution in another position over the evolutionary timescale. However, the major drawback of these approaches is that a large number (e.g. the order of 1,000) of homologous and non-redundant protein sequences are required to learn the underlying statistical model [204, 35]. Also, regarding studies on protein families using Evolutionary Tracing (ET) [124], the presence or absence of certain clusters of residue on a protein sequence is a main cause of divergence between globally-specific functions and family-specific functions [129]. Mutagenesis data is required for their studies, and their results suggest that the presence or absence of the co-occurring patterns is likely to be linked up with functional divergence [129].

Chapter 3

Aligned Pattern Clusters

3.1 Chapter Introduction

In this study, we present the Aligned Pattern Synthesis Process (AP Synthesis Process) which searches, aggregates, and aligns similar patterns from discovered patterns. First we use a sequence pattern discovery algorithm [220] that discovers and prunes a set of statistically significant non-redundant patterns and then aligns and clusters them. By statistical significance, we mean an imposed statistical criterion such that the association pattern must significantly deviate from its default random variable with identical and independently distribution under the null hypothesis. By a redundant pattern, we mean that the pattern is already covered by a super-pattern that contains it, or the calculated statistical significance is contributed by the presence of the strong statistically significant sub-patterns it contains. In this algorithm, first we take a set of multiple sequences of a

protein family as input. Then, we discover and locate the statistically significant amino acid association patterns, in linear time and space, while pruning redundant patterns based on the methodology we have previously developed [220].

Next, we present a new algorithm that aligns and clusters the discovered patterns into what we call Aligned Pattern Clusters (AP Clusters). We use an hierarchical clustering algorithm coupled with a dynamic programming alignment procedure with similarity scores and termination conditions to obtain AP Clusters. We then rank them according to their statistical significance. The rationale behind aligning and clustering patterns is that once an AP Cluster with its relative position is obtained, it will reflect the statistically significant residue association in the patterns (with variations) and also the amino acid distribution of each of its aligned columns to reveal the functionality of the protein family within the regions spanned by the patterns in the AP Clusters with statistical ranking and support. AP Clusters represent protein functional patterns, specifically binding segments, wherever they are in the input sequences of the protein family.

Applying our AP Synthesis Process to the cytochrome c, ubiquitin, and triosephosphate isomerase (TIM) protein families, we found that the AP Clusters do correspond to the functional binding segments that contain binding residues in all three protein families. The cytochrome c protein covalently binds the heme [42] attached to two cysteine residues. The heme's iron ion is chemically bonded to two binding residues from the opposite sides of the protein, each of them is surrounded by a sequence pattern with variations (i.e. within the discovered AP Cluster) referred to as the binding segment. Similarly, the ubiquitin protein contains seven lysine amino acids as binding residues that function by linking individual ubiquitin to create unique poly-ubiquitin recognized by different ubiquitin binding

proteins. Again, each of them is surrounded by a sequence pattern with variations as the binding segment. The AP Clusters found in TIM cover both functionally and structurally important binding sites to the ketose (DHAP) and aldose (GAP) substrates, which are transformed from one to the other through catalysis. In each of the protein families, we discover AP Clusters covering significant binding sites. In addition, the AP Synthesis Process runs faster than other motif finding algorithms and is not restricted by parameters such as fixed length and number of variations. This dissertation chapter is organized as follows: this section on Methods describes the proposed methodology; this section on results provides the *in silico* experimental results as evidence of the effectiveness of the proposed algorithm; and the last results compare different methodologies against our strong, weak, and conserved AP Clusters.

3.2 Methods

The Input Sequences

Let Σ be an alphabet containing the set of elements $\{\sigma_1, \sigma_2, \ldots, \sigma_{|\Sigma|-1}, \sigma_{|\Sigma|}\}$. As an example, the English alphabet contains 26 characters, {'a', 'b', ..., 'y', 'z'}= Σ, mathematically, σ_1 ='a', σ_2 ='b', ..., σ_{25} ='y', σ_{26} ='z', and $|\Sigma| = 26$.

A Set of Multiple Sequences Let $\mathbb{S} = \{s^k | k = 1, ..., |\mathbb{S}|\} = \{s^1, s^2, \ldots, s^{|\mathbb{S}|-1}, s^{|\mathbb{S}|}\}$ be the set of multiple sequences that represents the set of input sequences, where $|\mathbb{S}|$ is the total number of input sequences, and each sequence has length $|s^1|$, $|s^2|$, ..., $|s^{|\mathbb{S}|-1}|$, $|s^{|\mathbb{S}|}|$

respectively.

Note that the input sequences is also called the data space. Let each sequence, say sequence k, be $s^k = s_1^k \dots s_j^k \dots s_{|s^k|}^k$, where $s_j^k \in \Sigma$ is the element found in sequence k at position j of that particular sequence. Together the data space is the set of sequences composed of consecutive elements taken from the alphabet Σ as

$$
\begin{aligned}
s^1 &= s_1^1 s_2^1 s_3^1 \dots s_{|s^1|}^1, && (3.1)\\
s^2 &= s_1^2 s_2^2 s_3^2 \dots s_{|s^2|}^2, && (3.2)\\
&\vdots && (3.3)\\
s^k &= s_1^k \dots s_j^k \dots s_{|s^k|}^k, && (3.4)\\
&\vdots && (3.5)\\
s^{|\mathbb{S}|} &= s_1^{|\mathbb{S}|} s_2^{|\mathbb{S}|} s_3^{|\mathbb{S}|} \dots s_{|s^{|\mathbb{S}|}|}^{|\mathbb{S}|}, && (3.6)
\end{aligned}
$$

A Single Sequence Let s^k be a sequence indexed by k composed of consecutive elements taken from the alphabet Σ. $s^k = s_1^k s_2^k \dots s_{|s^k|-1}^k s_{|s^k|}^k$, where each $s_i^k \in \Sigma$ and s^k is of length $|s^k|$. For example, `bdxejrtewkwkHELLOkcmstsjavtpi` is a sequence of length 29. In this example the pattern is capitalized for the convenience of the reader so it could be easily observed. This sequence is represented by s^1, where $|s^1| = 29$, and the character at position 13 is $s_{13}^1 =$`H`.

Definition 1 *Each input sequence s^k has a class label y^k, i.e., (s^k, y_k). Let $y = \{y_1, y_2, ..., y_{|S|}\}$ be the set of class label corresponding to the set of sequences indexed by k, where each $y^k \in Y = \{class1, class2, ..., class_|Y|\}$, which are a set of class names.*

3.2.1 The Pattern Discovery Step

In the Pattern Discovery Step, we apply our pattern discovery and pattern pruning algorithm [220] that uses a linear time and space suffix tree to obtain a condensed list of significant patterns from the family of protein sequences.

Pattern Discovery Definitions

Definition 2 *A set of unaligned patterns is defined as* $\bar{\mathbb{P}} = \{\bar{p}^i | i = 1, ..., |\bar{\mathbb{P}}|\} = \{\bar{p}^1, \bar{p}^2, \ldots, \bar{p}^{|\bar{\mathbb{P}}|-1}, \bar{p}^{|\bar{\mathbb{P}}|}\}$ *corresponding to a resulting set of Aligned Patterns* $\mathbb{P} = \{p^i | i = 1, ..., |\mathbb{P}|\} = \{p^1, p^2, \ldots, p^{|\mathbb{P}|-1}, p^{|\mathbb{P}|}\}$ *of a fixed length elaborated in the definition for the AP Cluster. An unaligned pattern* $\bar{p}^i = s^i_1 s^i_2 ... s^i_{|\bar{p}^i|}$ *is an exact substring from* $\mathbb{S}$ *that passes four statistical conditions refined to a score defined by Wong et al.[220].*

An occurrence of the pattern $\bar{p}^i$ is expressed as $occ(\bar{p}^i) = j_i$ such that $\bar{p}^i = s^i_{j_i} s^i_{j_i+1} \cdots s^i_{j_i+|\bar{p}^i|-1}$, where i is the index of the sequence in which that pattern occurs, and j_i is the starting index in that sequence s^i where the pattern begins.

$$s^1 = s^1_1 ... \boldsymbol{s^1_{j_1+1} s^1_{j_1+2} ... s^1_{j_1+|\bar{p}^i|-1} s^1_{j_1+|\bar{p}^i|}} ... s^1_{|s^1|} \tag{3.7}$$

$$s^2 = s^2_1 ... \boldsymbol{s^2_{j_2+1} s^2_{j_2+2} ... s^2_{j_2+|\bar{p}^i|-1} s^2_{j_2+|\bar{p}^i|}} ... s^2_{|s^2|} \tag{3.8}$$

$$\ldots \tag{3.9}$$

$$s^{|\mathbb{S}|} = s^{|\mathbb{S}|}_1 ... \boldsymbol{s^{|\mathbb{S}|}_{j_{|\mathbb{S}|}+1} s^{|\mathbb{S}|}_{j_{|\mathbb{S}|}+2} ... s^{|\mathbb{S}|}_{j_{|\mathbb{S}|}+|\bar{p}^i|-1} s^{|\mathbb{S}|}_{j_{|\mathbb{S}|}+|\bar{p}^i|}} ... s^{|\mathbb{S}|}_m \tag{3.10}$$

$$\tag{3.11}$$

Here, a text example (Table 5.1) is examined in detail and presented for clearer understanding of the cluster validity measures. The class labels adopted here have no functional meaning as those related biological classes; they are just class names. These dataset (Table 5.1) contains three functional patterns of the English words, `HELLO`, `MELLOW`, and `BELLOW`, which are embedded in fifteen multiple sequences, associated with three class characteristics: happy, sad, angry.

Table 3.1: Example of Patterns $\bar{p}^1$ =`HELLO`, $\bar{p}^2$ =`MELLOW`, and $\bar{p}^3$ =`BELLOW`

$\mathbb{S}$	The Input Sequences
s^1	bdxejrtewkwkHELLOkcmstsjavtpi
s^2	nfixtHELLOuzdovcaaxnkjfjcvwk
s^3	dimtndvkjmkHELLObkcmstsj
s^4	tzhgarzofdHELLOpwkxmc
s^5	tyjxjqnyHELLOwmopemlqfgptnwnq
s^6	kntywtoaxMELLOWbtiasycma
s^7	jilxchitivMELLOWriiiweyfzgvuyaa
s^8	hmlzvMELLOWorgfeb
s^9	xhmlzvqgcanyMELLOWgbfj
s^{10}	vqgcanyffcMELLOWvcnsnjvalbdvr
s^{11}	cbpyhejgkinrphceBELLOWndwzahvkitagtt
s^{12}	ndwlofBELLOWscktbucwqnboeaaklknsrmur
s^{13}	fzomphnlrqhupkqBELLOWyutpfu
s^{14}	skwybrfiBELLOWyvxjdijwqjvs
s^{15}	nknhqexqieaBELLOWybnvrhpnsjnfms

Definition 3 *Let $\mathbb{D}(\bar{p}^i)$ be all the occurrences of the pattern, $\bar{p}^i$, found in the input sequence. We refer to $\mathbb{D}(\bar{p}^i)$ as the data induced by $\bar{p}^i$ or the induced data of $\bar{p}^i$. $\mathbb{D}(\bar{p}^i)$ will later be used to compute the cluster validity measures to reveal how many amino acids in each aligned column in an AP Cluster that may correspond to protein classes.*

Pattern Discovery Algorithm: Statistical Conditions for Discovering a Pattern

The Pattern Discovery Step takes advantage of a fast and space-efficient algorithm to discover high-order patterns that are statistically significant and not redundant [220]. Existing pattern discovery algorithms use statistical conditions as confidence thresholds to restrict the patterns discovered. Two existing statistical conditions are frequency count (Table 3.2 (A)), which discovers FREQUENT patterns, and the standard residual test (Table 3.2 (B)), which discovers STATISTICALLY SIGNIFICANT patterns against the random background model that is identically and independently distributed. To remove redundant patterns, Wong *et al.*[220] introduced a pattern pruning algorithm built into the pattern discovery algorithm with two additional statistical conditions: (1) DELTA-CLOSED (Table 3.2 (C)), and (2) STATISTICALLY NON-INDUCED (Table 3.2 (D)). With condition (1), the algorithm removes those redundant patterns that are already represented by their super-patterns; with condition (2) it prunes statistically significant patterns that are actually induced by their strong statistically significant sub-patterns. Details of these definitions are found in Wong et al. [220] and the four statistical conditions are presented in Table 3.2, where P is the pattern, P' is the super-pattern, and P'' is the sub-pattern. The Pattern Discovery

Table 3.2: Four Statistical Conditions

Conditions	Existing	Not Redundant
(1) Frequency Count	(A) FREQUENT $count(P) > c$	(C) DELTA-CLOSED $\frac{count(P')}{count(P)} < \delta$
(2) Standard Residual	(B) STATATISTICALLY SIGNIFICANT $z_P \geq t$	(D) NON-INDUCED $z_{P\|P''} \geq t$

Step with the four statistical conditions from Table 5.1 is executed on the text example. Each of the resulting patterns and its corresponding four statistical conditions are listed in Table 3.3. Each column is a statistical condition from Table 3.2 and the patterns satisfy the conditions.

Table 3.3: Example of the Pattern Discovery Step

	(A) c	(B) z_P	(C) δ	(D) $z_{P\|P''}$
H E L L O	5	904.06	0.8	904.06
M E L L O W	5	5917	0.8	13.97
B E L L O W	5	5917	0.8	13.97

3.2.2 The Aligned Pattern Clustering Step

For the AP Clustering Step, we use a previously developed pattern clustering algorithm to produce a condensed list of AP Clusters that is flexible in entropy with respect to coverage [116]. The algorithm groups a set of similar patterns of different lengths obtained from the Pattern Discovery Step while simultaneously assembling them into aligned sets of patterns of the same length by inserting gaps and wildcards. The amino acids amongst the patterns are aligned in the same site (aligned column), reflecting its regional functionality within the sequence.

Align Pattern Clustering Definitions

Definition 4 *A set of AP Clusters* $\mathbb{C} = \{C^l | l = 1, \ldots, |\mathbb{C}|\} = \{C^1, C^2, \ldots, C^{|\mathbb{C}|-1}, C^{|\mathbb{C}|}\}$ *An AP Cluster, represented by* C^l*, is a group of similar patterns that have been optimally*

grouped and vertically aligned into a set of patterns $\mathbb{P}^l = \{p^1, p^2, \ldots, p^m\}$*,and is expressed as*

$$C^l = \texttt{ALIGN}(\mathbb{P}^l), \tag{3.12}$$

$$= \begin{pmatrix} s_1^1 & s_2^1 & \ldots & s_n^1 \\ s_1^2 & s_2^2 & \ldots & s_n^2 \\ \vdots & \vdots & \vdots & \vdots \\ s_1^m & s_2^m & \ldots & s_n^m \end{pmatrix}_{m \times n} = \begin{pmatrix} p^1 \\ p^2 \\ \vdots \\ p^m \end{pmatrix}, \tag{3.13}$$

$$= \begin{pmatrix} c_1 & c_2 & \ldots & c_n \end{pmatrix}. \tag{3.14}$$

where $s_j^i \in \Sigma \cup \{-\} \cup \{*\}$ *is a pattern* p^i *with a newly aligned column index* j*. Each of the* $|\mathbb{P}^l| = m$ *patterns in the rows of* C^l *is of length* $|C^l| = n$*.*

For the text example, the AP Clustering Step creates an AP Cluster containing three patterns with six aligned columns (Table 3.4).

Table 3.4: Example of an AP Cluster for the text example

$p^i \setminus c_j$	$\begin{pmatrix} c_1 & c_2 & c_3 & c_4 & c_5 & c_6 \end{pmatrix}_{1\times 6}$
$\begin{pmatrix} p^1 \\ p^2 \\ p^3 \\ p^4 \\ p^5 \\ p^6 \end{pmatrix}_{6\times 1}$	$\begin{pmatrix} H & E & L & L & O & * \\ B & E & L & L & O & W \\ M & E & L & L & O & W \\ B & A & L & L & S & * \\ B & A & L & K & S & * \\ H & A & L & \_ & S & * \end{pmatrix}_{6\times 6}$

Definition 5 *An Aligned Pattern, which will simply be referred to as a pattern from this*

point forward, is a subsequence of order-preserving elements maximizing the similarity of the patterns against a set of patterns from AP Cluster, $\mathbb{P}^l$ *of size* $|\mathbb{P}^l| = m$ *with gaps, wildcards (any amino acid from the protein alphabet), and mismatches to the length* $|C^l| = n$*. Let* $p^i = s^i_1 s^i_2 \dots s^i_{|p^i|}$*, where* $s^i_j \in \Sigma \cup \{-\} \cup \{*\}$ *is an pattern* p^i *with a newly aligned column index* c_j*.*

Definition 6 *An aligned column* c_j *in* C^l *represents the* j^{th} *horizontal position of amino acids from the set of patterns that forms the current AP Cluster,* $C^l = \begin{pmatrix} c_1 & c_2 & \dots & c_n \end{pmatrix}$.

A conserved column is an aligned column that is conserved to only one type of amino acid such that $c_j = [\sigma \dots \sigma \dots \sigma]^T$ where $\sigma \in \Sigma$.

Next, we identify the subset of distinct amino acids that comprises a particular aligned column, which will be used to calculate its cluster validity measures in Chapter 5. Since an AP Cluster is composed of a set of aligned sequence patterns, an amino acid in the j^{th} aligned column, c_j, of the patterns in the AP Cluster. Hence, we identify the amino acids on an aligned column via the patterns in the AP Cluster.

Definition 7 *Let* $\Sigma(c_j)$ *be the set of distinct amino acids that are restricted by an aligned column* c_j*. In addition,* σ *is also restricted by the entire pattern it occurs on:*

$$\Sigma(c_j) \quad = \quad \{\sigma = s^i_j | p^i = s^i_1 ... s^i_j ... s^i_n, p^i \in \mathbb{P}^l, \sigma \in \Sigma\}.$$

We denote $\sigma(c_j)$ *as an amino acid in* $\Sigma(c_j)$*, i.e.,* $\sigma(c_j) \in \Sigma(c_j)$*. The notation is used as* $pr(\sigma(c_j))$*, which means* $pr(c_j = \sigma)$*.*

In the text example, the pattern for the third row is $p^3 = HELLO$, and the aligned column for the first position is $c_1 = [HBM]^T$. The set of amino acids in the aligned column c_1 is $\Sigma(c_1) = \{H, B, M\}$, and the set of amino acids in the aligned column c_6 is $\Sigma(c_6) = \{*, W\}$.

Definition 8 *Let $\mathbb{D}^l$ be the data induced by AP Cluster C^l, which is the subset of the input sequences, or data space, that is caused by all the occurrences of the patterns contained in the AP Cluster, $C^l = \{p^1, p^2, \ldots, p^m\}^T$. We identify $\mathbb{D}^l$ the data induced by C^l or the induced data of C^l. As a result, $\mathbb{D}^l$ is the union of the extended data (or input sequences) induced by all the patterns contained in C^l, $\mathbb{D}^l = \mathbb{D}^1 \cup \mathbb{D}^2 \cup \cdots \cup \mathbb{D}^m$.*

The Overall Align Pattern Clustering Algorithm

The AP Clustering Step is accomplished by the single-linkage hierarchical clustering algorithm that takes an input of a list of patterns and then synthesizes, or more precisely, aligns and groups them into one or more AP Cluster(s) (Algorithm 1). We modified a hierarchical clustering algorithm that synthesizes random sequences [39, 213]. The bi-clustering nature of the hierarchical clustering algorithm allows sub-clusters to be analyzed with ease in polynomial time complexity. The hierarchical clustering algorithm iteratively merges two AP Clusters in a pairwise-manner based on their similarity scores until one of the termination conditions is reached. The three key parameters of the algorithm are the MERGE Algorithm, the SIMILARITY Score, and the TERMINATION Condition. Using the text example, Fig. 3.1 demonstrates one iteration of the hierarchical clustering algorithm. More precisely, it shows the last step of the iterative MERGE between AP Cluster C_1 and

Algorithm 1 The Single-Linkage Hierarchical Clustering Algorithm

Require: $\mathbb{P} = \{\bar{P}_1, ..., \bar{P}_{|\mathbb{P}|}\}$, where $|\mathbb{P}| = m$
Ensure: $\mathbb{C} = \{C_1, ..., C_{|\mathbb{C}|}\}$
Set all $P_i \in \mathbb{P}$ as $C_i \in \mathbb{C}$
while (For all pairs of clusters $(C_i, C_j) \in \mathbb{C}$) **do**
    Calculate SIMILARITY(C_i, C_j)
end while
while (! TERMINATION Conditions) **do**
    Select max SIMILARITY(C_{max_i}, C_{max_j})
    MERGE$(C_{max_i}, C_{max_j}) = C_{new}$
    Update list of clusters $\mathbb{C}$
    while (For all pairs of clusters (C_{new}, C_i)) **do**
        Calculate SIMILARITY (C_{new}, C_i)
    end while
end while

AP Cluster C_2, thereby creating the new AP Cluster C_3.

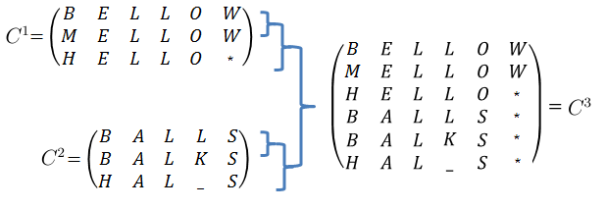


Figure 3.1: The last step in hierarchical clustering. In one iterative step of hierarchical clustering, an existing AP Cluster, C_1 is merged with another AP Cluster, C_2, to result in the new AP Cluster, C_3.

Theoretical Runtime Complexity by Big-O Analysis For the runtime of our AP Synthesis Process (Table 3.5), we assume that m is the number of discovered patterns and n is the number of aligned columns of the AP Cluster. The first Pattern Discovery Step results in m number of patterns in $O(N)$ time, where N is the total input size as described

in Wong *et al.* [220]. Next, the AP Clustering Step synthesizes a set of AP Clusters as described in The Single-Linkage Hierarchical Clustering Algorithm. The initiation of each pattern from line 1 is $O(m)$ time and each pair of AP Clusters afterwards, from lines $2-4$, needs $O(m^2)$ time. The main portion of the hierarchical clustering algorithm is a loop from lines $5-12$ that halts when the TERMINATION Conditions are satisfied. In the worst case scenario, the loop in line 5 executes m times, when only one pattern is merged into the main cluster at each iteration. In the best case scenario, the loop executes $\lceil log(m) \rceil$ times, where two evenly sized AP Clusters are merged at each iteration. Moreover, the TERMINATION Conditions typically halt the loop earlier in even fewer iterations. Line 6 selects the maximum value in only $O(|\mathbb{C}|)$ time, which is not the longest limiting runtime step in the loop. In line 7, the main dynamic programming algorithm of the MERGE Algorithm is $O(m^2)$ time, when using the SIMILARITY Score to score each column. The sum-of-pairs scores require an additional $O(m_1 m_2)$ time and the entropy scores need an additional $O(m_1 + m_2)$ time, where m_1 and m_2 are the number of patterns in the first and the second AP Clusters, respectively. Lastly, line 9 compares the newly created AP Cluster with all the other existing clusters in $O(m)$ steps. Thus, the overall time complexity of algorithm is $O((m_1 m_2)m^2 n^2)$ or $O((m_1 + m_2)m^2 n^2)$. Note that the $O(m^4)$ is with respect to m, which is the total number of patterns discovered in the Pattern Discovery Step. Therefore to control the runtime for large datasets, the parameters to the Pattern Discovery Step can be restricted to limit the number of discovered patterns that are inputted into AP Clustering Step.

In addition, turning on the overlapping support set as a TERMINATION Condition is a computationally intensive threshold, where the support of the entire AP Cluster is

Table 3.5: Theoretical Runtime Calculations

	Runtime
Pattern Discovery Step	$O(N)$
AP Clustering Step	$O(n^2)$
Total	$O((m_1 m_2)m^2 n^2)$ $= O(m^4)$

computed. The loop executing the MERGE Algorithm halts if at least one overlapping support is found in the newly created AP Cluster. The support for each pattern in the AP Cluster is aggregated and then checked for overlap. This TERMINATION Condition takes $O(|\mathbb{P}|numSequences) = O(m|\mathbb{S}|)$ time.

Comparison of Runtimes The following runtime comparison demonstrates that our AP Synthesis Process is faster than existing motif finding methods. The experimental runtime was recorded for three protein families: cytochrome c, ubiquitin, and TIM (Table 3.6). We observed that our AP Synthesis Process was substantially faster than the other motif finding methods. It should be noted that all the other algorithms were executed under their default settings. Our AP Synthesis Process is faster because the noisy sequence variations with weak statistical support were not discovered and thus did not pass as a list of input patterns to the AP Clustering Step. Overall, the input consists of a shortened list of discovered patterns rather than the original input sequences. Thus, the runtime for synthesizing patterns is faster than that for synthesizing all input sequences due to the smaller search space. Also, in the next AP Clustering Step, the pairwise comparisons of our hierarchical clustering algorithm are faster than the full, all-way comparisons of full-linkage k-means clustering algorithms.

Table 3.6: Runtime Comparisons in Seconds

Methods	Cyto c	Ubi	TIM
AP Cluster	**0.18**	**0.04**	**0.16**
qPMS7 (length$\leq$ 10) [54]	0.21	0.18	0.83
qPMS7 (length$>$ 10) [54]	16.49	17.98	20.16
BLOCKS [77]	2.53	0.15	0.46
Gibbs (w=8-16) [109]	3.32	1.12	2.99
PROJECTION [32]	99.20	4.82	0.25
MEME [14]	111.01	13.98	43.38
CONSENSUS (L=8-16) [80]	289.38	13.53	15.79

The Merge Algorithms

The MERGE Algorithm iteratively merges two AP Clusters into one during hierarchical clustering. Two possible alignment algorithms are considered in this study: the global Needleman-Wunsch alignment algorithm [136] and the local Smith-Waterman alignment algorithm [174]. An alignment algorithm is essentially a dynamic programming algorithm with two steps: (1) forward-scoring that builds a score table by optimizing the sub-scores recursively and (2)back-tracking that steps through the score table in reverse from the optimal score to the first possible score in order to arrive at the final solution. The runtime for computing the score table of two AP Clusters, C_1 and C_2, in the dynamic programming algorithm is $O(|C_1||C_2|)$. Note that, depending on the type of SIMILARITY Score selected, a linear time complexity is added as described in the next section.

In the resulting score table for the dynamic programming, the final SIMILARITY Score of the new AP Cluster that is computed from two existing AP Clusters, represented by S_{AP}, is used to select the pair of AP Clusters to MERGE at each iteration of the hierarchical clustering process. A reward and a penalty are used to calculate the total score of an

AP Cluster, S_{AP}. A reward, S_{col}, is added for matching the amino acids in two aligned columns of each original AP Cluster; a penalty score is deducted for gaps. The equation below calculates the S_{AP} using global alignment, where c_i is an aligned column for C_1 and d_j is an aligned column for C_2.

$$S_{AP}[i,j] = max \begin{cases} S_{AP}[i-1,j-1] & +S_{col}(c_i,d_j), \\ S_{AP}[i,j-1] & +GapPenalty(-,d_j), \\ S_{AP}[i-1,j] & +GapPenalty(c_i,-). \end{cases} \tag{3.15}$$

The following equation calculates the S_{AP} using local alignment:

$$S_{AP}[i,j] = max \begin{cases} 0 & \\ S_{AP}[i-1,j-1] & +S_{col}(c_i,d_j), \\ S_{AP}[i,j-1] & +GapPenalty(-,d_j), \\ S_{AP}[i-1,j] & +GapPenalty(c_i,-). \end{cases} \tag{3.16}$$

In the resulting AP Cluster, the symbol '_', the gap, is used to represent the opening of the pattern by adding an empty null character. The symbol '*', the wildcard, is used to pad the beginning and end of patterns to represent any amino acid that is not a part of the pattern.

The Similarity Scores

Two major categories of SIMILARITY scores, the sum-of-pairs scores and the entropy-based scores, are examined for mismatches between two original AP Clusters in the MERGE Algorithm. The two aligned columns from each AP Cluster are combined to compute the S_{col} score. The sum-of-pairs scores have the runtime of $O(m_1|C_1|m_2|C_2|)$, and the entropy-based scores have the runtime of $O((m_1+m_2)|C_1||C_2|)$, where m_1 is the number of patterns in the first AP Cluster, C_1, and m_2 is the number of patterns in the second AP Cluster, C_2. To give a formal definition for each of the scores, first let $c_i = \begin{bmatrix} c_i^1 & c_i^2 & \dots & c_i^{m_1} \end{bmatrix}^T$ be an aligned column for C_1 and $d_j = \begin{bmatrix} d_j^1 & d_j^2 & \dots & d_j^{m_2} \end{bmatrix}^T$ be an aligned column in C_2, where each $c_i^k, d_j^l \in \Sigma$. The sum-of-pairs scores from the two aligned columns compare all pairs of amino acids by scoring each comparison as S_{one}, which is then summed to S_{col}.

$$S_{col}(c_i, d_j) = \sum_{\forall c_i^k \in c_i} \sum_{\forall d_j^l \in d_j} S_{one}(c_i^k, d_j^l). \tag{3.17}$$

One possible S_{one} is the Hamming distance, which satisfies the metric properties and thus can be summed. The matches are rewarded, and the mismatches and the gaps are penalized. We adapted a weighted Hamming distance in order to penalize weighted mismatches and weighted gaps differently. Table 3.7 presents the different S_{one} values and their weightings, where w is the weighting on the scores. We also used the BLOSUM[79]/PAM[50] substitution matrix to reward matches and penalize mismatches based on observed rate of change.

Alternatively, the entropy-based scores use the probability distribution of the existing amino acids occurring at the combined aligned columns. The two different entropy-based

Table 3.7: Four Possible S_{one} for the sum-of-pairs Scores

Score S_{one}	Match	Mismatch	Gap Penalty
Hamming Distance	$+1$	-1	-1
Weighted Gap	$+1$	-1	$-w$
Weighted Mismatch	$+w$	$-w$	-1
BLOSUM/PAM	matrix	matrix	-1

scores considered are:

- *Information Entropy Score*

$$S_{col}(c_i, d_j) = H(c_i \cup d_j) \tag{3.18}$$

$$= -\sum_{\sigma \in c_i \cup d_j} Pr(\sigma) \log Pr(\sigma), \tag{3.19}$$

  where $Pr(\sigma)$ is the probability distribution of $\sigma \in \Sigma$ from the combined aligned columns, $c_i \cup d_j$.

- *Information Gain Score* [39]

$$S_{col}(c_i, d_j) = w_1 H(c_i) + w_2 H(d_j) - H(c_i \cup d_j), \tag{3.20}$$

  where n_1 is the size of c_i and n_2 is the size of d_j such that $w_1 = \frac{n_1}{(n_1+n_2)}$ and $w_2 = \frac{n_2}{(n_1+n_2)}$.

Returning to the text example, consider the last step of the MERGE preformed on AP Cluster C_1 and AP Cluster C_2 using the Global Alignment as the MERGE Algorithm and

the Hamming Distance as the SIMILARITY Score. The resulting dynamic programming score table for C_1 and C_2 is illustrated in Table 3.8. Each cell contains the value of the score with an arrow indicating the backtrace position in the clusters. The negative values in each of the cells of the score table are the SIMILARITY Score. The optimal solution is marked with a '*' as it is the backtracked solution, shown by the corresponding arrows through the score table. All entropy scores were normalized and scaled to take on values between -1 and $+1$ in order to reward matches and penalize mismatches, thereby adjusting the entropy by offsetting and scaling the final entropy value. Because the AP Cluster becomes more random at each iteration, the negative penalty of -1 causes the entropy scores to become more negative in the score table.

Table 3.8: The Score Table Combining Two Final AP Clusters using Dynamic Programming

				d_1		d_2		d_3		d_4		d_5	
				B		A		L		L		S	
				B		A		L		K		S	
				H		A		L		-		S	
c_1	B	M	H	*−0.740		−1.289	↖	−1.837	↖	−2.837	↖	−3.63	↖
c_2	E	E	E	−1.445	↖	*−1.740	*↖	−2.289	↖	−2.287	↖	−3.287	↑
c_3	L	L	L	−2.151	↖	−2.445	↖	*−0.740	*↖	−1.74	↖	−2.74	↑
c_4	L	L	L	−2.786	↖	−3.139	↖	−1.445	↖	*−0.827	*↖	−1.827	↑
c_5	O	O	O	−3.627	↖	−3.512	↖	−2.139	↖	−1.827	←	*−1.769	*↖
c_6	W	W	-	−4.438	↖	−4.333	↖	−3.139	←	−2.827	←	−2.769	←

The Termination Conditions

The TERMINATION condition of the MERGE Algorithms, just like the SIMILARITY Score chosen, also determines the quality of the final AP Clusters synthesized. The numerical

thresholds for TERMINATION Conditions considered are 1) the threshold on the value of the Average Cluster Entropy, 2) the total number of clusters, 3) the number of patterns in each cluster, and 4) the threshold on the percentage change in the SIMILARITY score. Lastly, Overlapping Support for the TERMINATION Condition, a non-numerical threshold, is also used, where the hierarchical clustering process is halted when the AP Cluster occurs more than once in a single sequence called overlapping support.

3.2.3 The AP Cluster Refinement Step

For the AP Cluster Refinement Step, we improve the sequence coverage while maintaining the entropy. We call the two types of refined AP Clusters the Weak AP Cluster and the Conserved AP Cluster. Each original AP Cluster, which will be referred to as the Strong AP Cluster, has a corresponding Weak AP Cluster as well as a corresponding Conserved AP Cluster. First we expand an AP Cluster to a Weak AP Cluster by finding the best matching occurrence for each of the remaining sequences not covered by the AP Cluster, and then removing those occurrences that are far away in relative position, which may be false positive occurrences. In this manner, outlying occurrences that do not match the pattern in the AP Cluster precisely can be covered; higher mutational variation allows more sequences to be covered by the corresponding Weak AP Cluster. Although the Weak AP Cluster increases the number of sequences covered, it also increases the entropy. Thus, we further refine the Weak AP Clusters to the Conserved AP Clusters by restricting the conserved columns from the original AP Clusters to reduce the number of sequences covered, and thus decrease the entropy; these are instances that satisfy the

Weak AP Cluster while adhering to the conserved column. In other words, the conserved amino acids in the variable pattern (i.e. conserved column) are required to be fixed, while additional variations in the other non-conserved aligned columns are allowed.

3.2.4 Artificial Datasets for Parameter Tuning

To test the experimental runtime and quality of the resulting AP Clusters of our method, we created nine sets of synthetic input data containing synthetic patterns of length 10, where each pattern occurs with a frequency of 5 and each pattern has a 10% chance of mutation at a random position from the previous pattern. Each dataset varies from the last amino acid and contains five occurrences of the synthetic pattern. The Pattern Discovery Step was executed with the following parameters: *minimal order*of 3, *confidence interval* of 3, *minimum occurrence* of 5, and *delta* of 0.8. The parameters of AP Clustering Step were MERGE *Algorithm* set as Global Alignment with SIMILARITY *Score* set as Hamming Distance and no TERMINATION *Condition*.

Runtime Comparison of Similarity Scores

To compare the runtime of our AP Synthesis Process for each of the SIMILARITY Scores, we plotted our experimental runtime using each of the MERGE Algorithms. The experimental runtime is measured by counting the number of character comparisons, which was plotted against the number of synthetic patterns in the dataset. The curves of five SIMILARITY Scores are plotted for both the Global and Local alignments (Fig. 3.2). The plotted runtime curve of our AP Synthesis Process is polynomial with respect to the number of

patterns in the cluster. As described in the SIMILARITY Score section, the sum-of-pairs scores performed $O(m)$ slower than the entropy scores due to a more complete pairwise comparison.

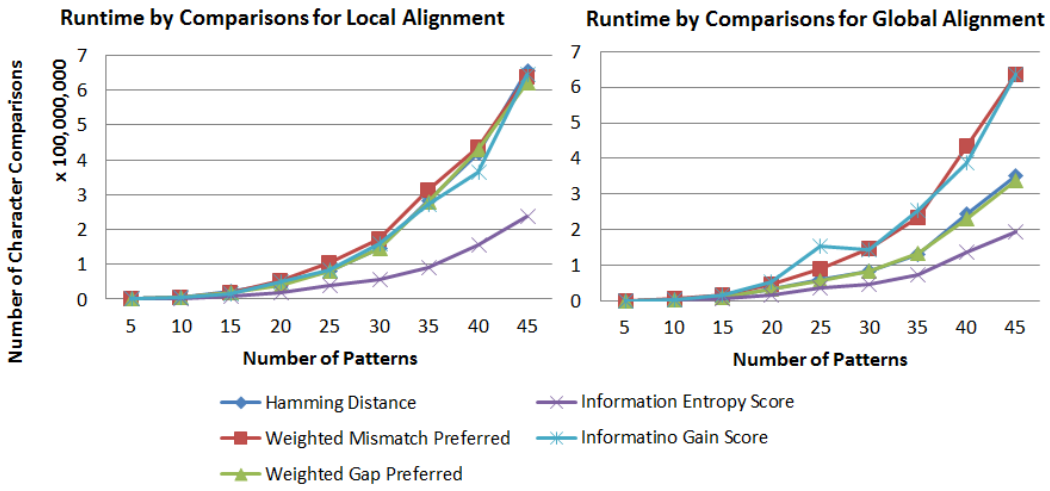


Figure 3.2: Runtime of Alignment Algorithms. The five SIMILARITY Scores are Hamming Distance, Weighted Mismatch Preferred, Weighted Gap Preferred, Information Entropy Score, and Information Gain Score. (a) The runtimes of the five SIMILARITY Scores are compared while executing the Local MERGE Algorithm. (b) The runtimes are compared while executing the Global MERGE Algorithm. The sum-of-pairs scores performed more slowly than the entropy scores, with the exception of Information Entropy Score due to the uneven sizes of the AP Clusters being merged.

Surprisingly, the Information Gain Score did not compute at $O(m^2)$ time as expected for Entropy Scores because the Information Gain Score causes a highly unbalanced cluster to be merged at each iteration.

Qualitative Comparisons of Similarity Score and Alignment Algorithm

To determine the parameters that yield the highest quality AP Clusters, we examined the combinations of the MERGE Algorithm with the SIMILARITY Scores. We measured the quality of the resulting AP Clusters using Average Cluster Quality, $Q(C)$, which is the inverse normalized information entropy of the aligned columns from all resulting AP Clusters:

$$Q(C) = \frac{\sum_{\forall C \in \mathbb{C}} \sum_{\forall c_j \in C} \sum_{\forall \sigma_i \in \Sigma} Pr(\sigma_i) \log Pr(\sigma_i)}{|C||\mathbb{C}|}, \tag{3.21}$$

where $H(c_j) = -\sum_{\forall \sigma_i \in \Sigma(c_j)} Pr(\sigma_i) \log Pr(\sigma_i)$, Σ is the alphabet, C is the aligned columns in the AP Cluster, and $\mathbb{C}$ is the set of resulting AP Clusters. When the $Q(C)$ is close to one, the resulting AP Clusters have a desirable quality, which is more stability. When the $Q(C)$ is close to zero, the resulting AP Clusters are more random.

The first set of tuning experiments identified the optimal combination of the MERGE Algorithm with the SIMILARITY Scores (Fig. 3.3). For the MERGE Algorithm, Global Alignment performs better than local alignment because it aligns the full pattern rather than the subpatterns of the pattern. Thus, for the five SIMILARITY Scores compared for Global Alignment, the sum-of-pairs scores performed better than the entropy scores because they exhaustively compare all pairs of amino acids from both aligned columns. These extra comparisons take $O(m)$ time longer to execute, where m is the number of patterns.

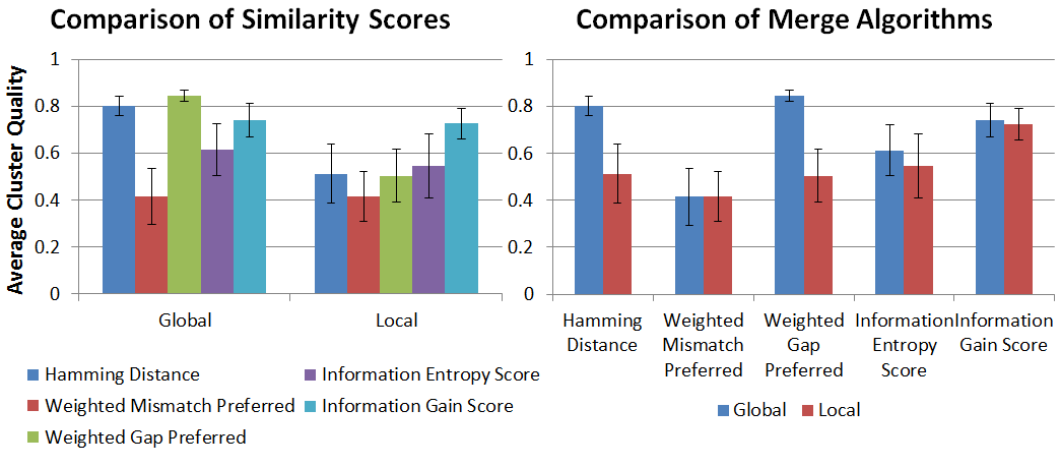


Figure 3.3: Tuning the MERGE Algorithm and SIMILARITY Score. (a) The ten $Q(C)$ are separated into the five SIMILARITY Scores. Of the two MERGE Algorithms compared, Global Alignment results in better AP Clusters, thus we focus on its scores. (b) The ten $Q(C)$ are separated into the two MERGE Algorithms. For Global Alignment, Hamming Distance performed the best.

The Termination Conditions

To identify the possible threshold values for each of the TERMINATION Conditions, we adjusted their values for each of the artificial datasets. Considering the results of the previous optimality experiments, we set the MERGE Algorithm to Global Alignment and the SIMILARITY Score to Hamming Distance, while varying the values of the TERMINATION Conditions and plotted the resulting $Q(C)$ as displayed in Fig. 3.4. The first TERMINATION Condition, the Number of Patterns per Cluster, results in an inverse exponential curve. Here, the ideal threshold value occurs before the quality of the AP Clusters begins to decrease rapidly. The second TERMINATION Condition fits a logarithmic curve because decreasing the number of clusters also increases the number of patterns, thereby increasing the randomness and decreasing the Average Cluster Entropy. The ideal threshold value

occurs before the curve levels off when the quality of the AP Cluster is rapidly increasing to the optimal value, which is one.

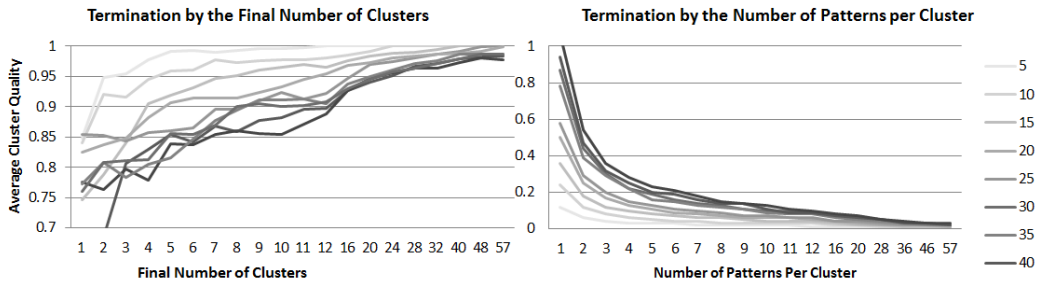


Figure 3.4: Threshold Trends of TERMINATION Condition. The two TERMINATION Conditions examined are (a) the Final Number of Clusters, which fits a logarithmic curve, and (b) The Number of Patterns per Cluster, which fits a inverse exponential curve.

3.3 *In Silico* Biological Experiments

We conducted a biological experiment on the cytochrome c and the ubiquitin protein families to examine how the resulting AP Clusters are related to the binding sites that associate with the most important functionality of the protein. There are three aspects we explored: the reduction of the set of candidate solutions from the discovered patterns to the AP Clusters obtained; how each pattern in the AP Cluster surrounding the binding site represents a binding segment in a single strand of protein; and how binding residues correlate to their column hyperedge. Finally, we display our results underneath the pFam multiple sequence alignment to compare the differences in the representations. In the comparison, we demonstrate the overall hierarchical clustering performance of our AP Synthesis Process as well as the quality of the resulting AP Clusters.

3.3.1 The Pfam Cytochrome C Protein Family

Cytochrome C Results

We confirmed that the binding segments of a protein family can be richly represented by AP Clusters. The 237 input sequences for the cytochrome c protein family were downloaded from Pfam (PF00034) on January 13th, 2010 from Pfam release 23. These Pfam seed sequences have an average length 94, identity 18%, and coverage 36.97%. Based on the identity and coverage of the protein family, we executed the Pattern Discovery Step with *minimum length* of 5, *delta* of 0.9, *confidence interval* of 3, and *minimum occurrence* of 10. To reduce the number of singular patterns, the minimum occurrence was adjusted up by two. We then executed the AP Clustering Step on the list of statistically significant patterns discovered by the Pattern Discovery Step. As concluded from our experiments with simulated data, the MERGE *Algorithm* was Global Alignment, the SIMILARITY *Score* was Hamming Distance, and the TERMINATION *Condition* was Non-Overlapping Supports.

Table 3.9 lists the statistically ranked patterns of the cytochrome c protein family resulted from the Pattern Discovery Step. All but one of these patterns corresponds to the proximal and distal binding residues (amino acids in bold), which are crucial for the binding functionality of the protein. By itself, each individual pattern with its variation has a low frequency count, which is a small fraction of the sequence support. Hence, a single pattern alone cannot represent the rich variation of the functional motif within the entire protein family. Therefore, the AP Cluster representing the binding sites, containing a set of similar patterns, that have been grouped and aligned with variations, provides a much richer description of the binding segments. In the AP Clustering Step, we demonstrate

that AP Clusters are able to richly capture the conservation and the variability of patterns in the aligned columns.

Table 3.9: Statistically Ranked Patterns from the Cytochrome C Protein Family

Ranking	Pattern	Frequency	Score	Binding Residue
1	CSMC**H**AREP	11	5021	His18
2	GRCSMC**H**A	11	928.8	His18
3	RCSMC**H**A	16	576.9	His18
4	MC**H**AREP	13	250.4	His18
5	SHA**M**PP	12	32.00	Met62
6	CAAC**H**G	10	19.68	His18
7	A**M**PPAN	12	18.27	Met62
8	IYLAG	10	12.59	
***9**	CAAC**H**	**22**	**27.97**	**His18**
10	CASC**H**	16	22.41	His18
***11**	**M**PLGN	**19**	**15.88**	**Met62**
12	HA**M**PP	16	12.94	Met62
13	CVAC**H**	12	12.32	His18
14	CAGC**H**	13	11.46	His18

The two highest ranking AP Clusters correspond to the proximal and distal binding segments of the cytochrome c protein family as displayed in Fig. 3.6. Tables 4.5 and 4.6 exhibit the proximal and distal AP Clusters, respectively, and their set of patterns and frequency counts from the Pattern Discovery Step. Once again, the binding residues crucial for the functionality of the protein family are represented by one single aligned column (in bold). The lower frequency of the distal pattern implies that the distal binding segment is not as well conserved as the proximal binding segment. This lower conservation is reflected by the lower statistical significance of the distal patterns contained in the distal AP Cluster. However, we were still able to identify the binding residue as a conserved column in the

distal AP Cluster.

Table 3.10: The Proximal AP Cluster of the Cytochrome C Family

c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9	c_{10}	c_{11}	Frequency
G	R	C	S	M	C	**H**	A	*	*	*	11
*	R	C	S	M	C	**H**	A	*	*	*	16
*	*	C	S	M	C	**H**	A	R	E	P	11
*	*	*	*	M	C	**H**	A	R	E	P	13
*	*	C	V	A	C	**H**	*	*	*	*	12
*	*	C	A	S	C	**H**	*	*	*	*	16
*	*	C	A	G	C	**H**	*	*	*	*	13
*	*	C	A	A	C	**H**	G	*	*	*	10
*	*	C	A	A	C	**H**	*	*	*	*	**22**
					C	**H**					**237**

Table 3.11: The Distal AP Cluster of the Cytochrome C Family

c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	Frequency
*	*	*	**M**	P	L	G	N	**19**
*	*	A	**M**	P	P	A	N	12
S	H	A	**M**	P	P	*	*	12
*	H	A	**M**	P	P	*	*	16
			M	P				**153**

First, we observed that the most frequent pattern in the proximal AP Cluster is pattern 9, 'CAACH', which covers 22 of the 238 sequences. The most frequent pattern in the distal AP Cluster is pattern 11, 'MPLGN', which covers 19 of the 238 sequences. By themselves, the frequency counts of the patterns are not strong enough to identify other conserved features around the binding residues. Therefore, the AP Cluster containing a set of similar patterns with variations provides a richer representation that summarizes the binding segment. Next, we defined a conserved column within the AP Cluster as an

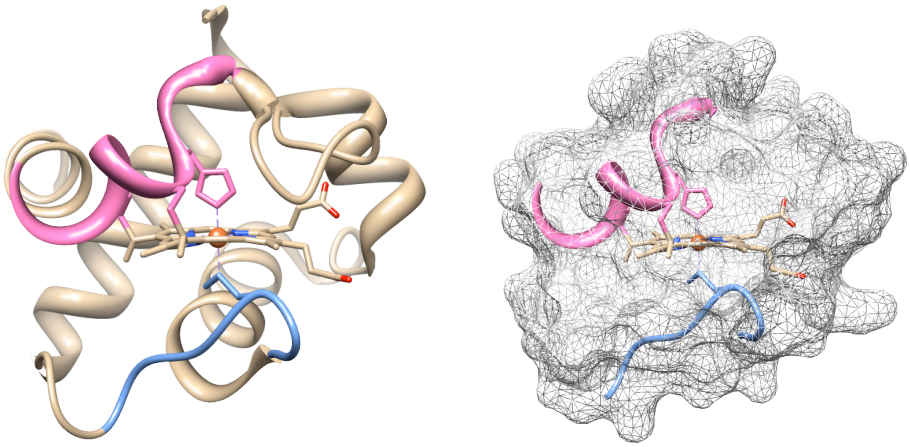
aligned column that has only one possible amino acid value amongst the patterns in the AP Cluster. The two conserved columns in the proximal AP Cluster are His18 and Cys17; similarly, the two conserved columns in the distal AP Cluster are Met62 and Pro63.

If we were to examine the pattern of the combined amino acids of [CH], this 2nd order pattern would occur in 237 of the 238 sequences, whereas the pattern of the combined amino acids of [MP] would occur in 153 out of the 238 sequences. Thus, the collection of conserved columns is able to reveal a strong low-order pattern that has less noise and true functional significance.

Cytochrome C Discussion

Biologically, the two binding residues in the cytochrome c protein are (1) the proximal binding residue [42, 100] and (2) the distal binding residue [178] (Fig. 3.5). Our study showed that these crucial binding segments correspond to AP Clusters that contain conserved columns, which are the binding residues, the main biological function of the protein.

The rows of the AP Clusters are aligned based on their horizontal patterns because of their statistical significance, and the aligned columns of the AP Clusters are grouped based on their vertical amino acid stability. To show the significance of the patterns, first, each AP Cluster contains a set of horizontal patterns that are similar to one another. Although these patterns suggest their horizontal significance in the protein sequences, individually, they do not identify the significance of the amino acid's conservation and variation. Thus, the stability of the aligned columns is important for the identification of the binding residues. Second, the aligned columns of each AP Cluster correspond to



(a) Cytochrome C Binding Segments (b) Cytochrome C Binding Residues

Figure 3.5: The 3D structure of cytochrome c (PDB ID: 1F1F). (a) The two binding segments represented as AP Clusters: the pink proximal binding segment and the blue distal binding segment. (b) Specifically, one particular amino acid from each of the AP Clusters binds the iron ion.

the conservation of the cluster, which otherwise is not easily identified in each individual non-variable pattern. The conserved columns of the AP Clusters correspond to binding residues. The two conserved columns in the proximal AP Cluster, His18 and Cys17, are essential to the functionality of the cytochrome c protein family for binding the heme ligand. More precisely, the His18 conserved column acts as the proximal binding residue, and the Cys17 conserved columns binds the thioether bond to the vinyl group on the heme. Similarly, the Met62 conserved column in the distal AP Cluster acts as the distal binding residue. Our proximal AP Cluster for cytochrome c is consistent with the proximal binding motif, [C]-x(2)-[CH], given by PROSITE (PDOC00169) [16, 172] and also with the strong emission probability from Pfam[175]. Moreover, our method identified the distal binding AP Cluster, which is not annotated by PROSITE and is identified by Pfam as

only a weak emission probability. Furthermore, our method also identified an additional conserved column, Pro63, in the distal binding segment which does not bind the heme ligand. Its role may be important for protein conformation during translation. The heme is attached to the cytochrome after translation; hence, the binding segments must permit conformational flexibility in order for the heme ligand to enter the binding pocket[75]. Since the proximal binding segment is a rigid secondary alpha-helix structure with three bonds to the heme ligand, the distal binding loop must be the flexible segment. We postulate that the Pro63 conserved column in the distal binding segment, which is a secondary loop structure, must bend to allow the binding site to open so that the heme ligand can enter during translation.

3.3.2 The Pfam Ubiquitin Protein Family

Ubiquitin Results

The input of ubiquitin is uniquely identified in Pfam by PF00240, which contains 78 seed sequences that have an average length of 67.1, identity 44%, and coverage 28.05%, was downloaded March 19th, 2012 from Pfam release 25. In this experiment, parameters were the same as cytochrome c, except for a *minimum occurrence* of 5 due to the protein family's sequence length, identity, and coverage.

In the Pattern Discovery Step, fourteen of the twenty-nine discovered patterns contain one of the seven binding residues. Those twenty-nine patterns are further compressed into eight AP Clusters with Average Cluster Score of 0.63. Protein motifs exhibit variability, and thus, AP Clusters represent the protein's binding sites more effectively and explicitly.

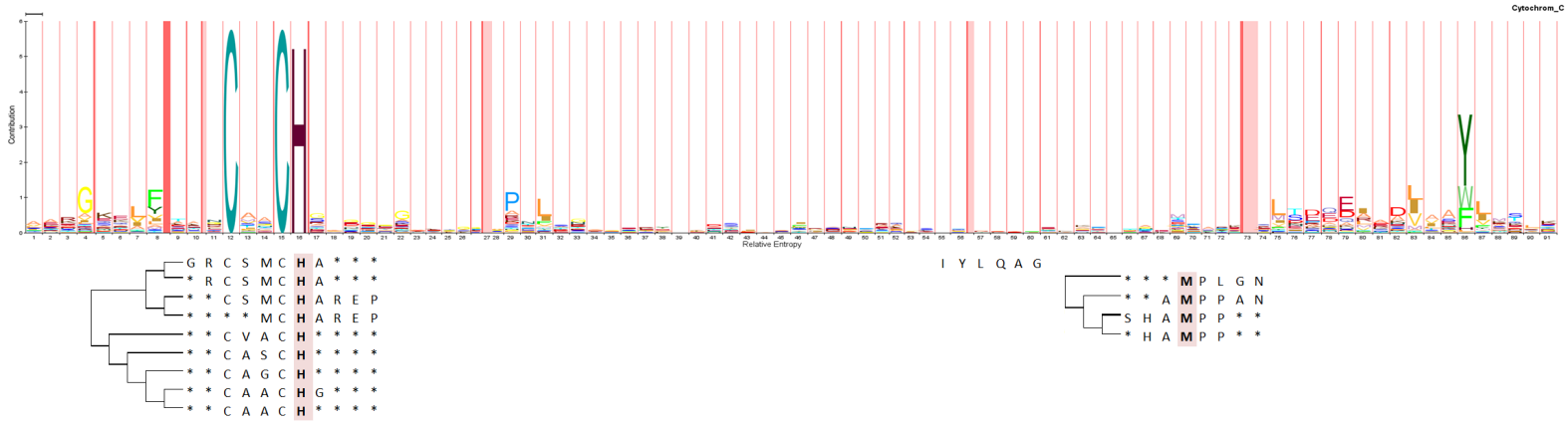


Figure 3.6: The HMM and AP Cluster comparison of cytochrome c. The two resulting AP Clusters in cytochrome c are compared to the HMM logo from Pfam. In the largest AP Cluster, Cys17 and His18 are identified as the conserved columns, where His18 binds the heme iron and Cys17 binds the heme. In the second largest AP Cluster, Met62 is the conserved column of the distal binding segment, where Met62 binds the heme iron.

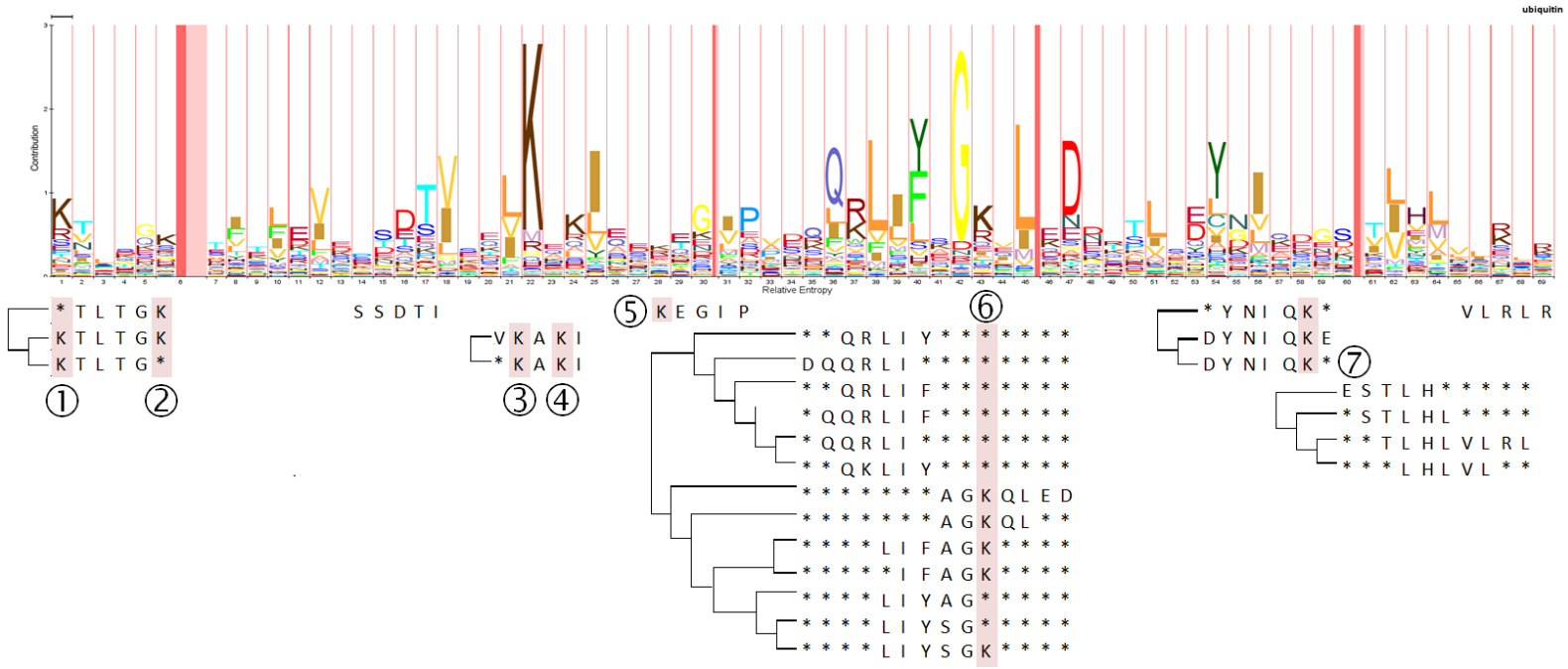


Figure 3.7: The HMM and AP Cluster comparison of ubiquitin. Our resulting eight AP Clusters are compared to the profile HMM logo from pFam. The seven Lys binding residues of the ubiquitin protein family are highlighted in red in the AP Cluster: Lys6, Lys11, Lys27, Lys29, Lys33, Lys48, and Lys63.

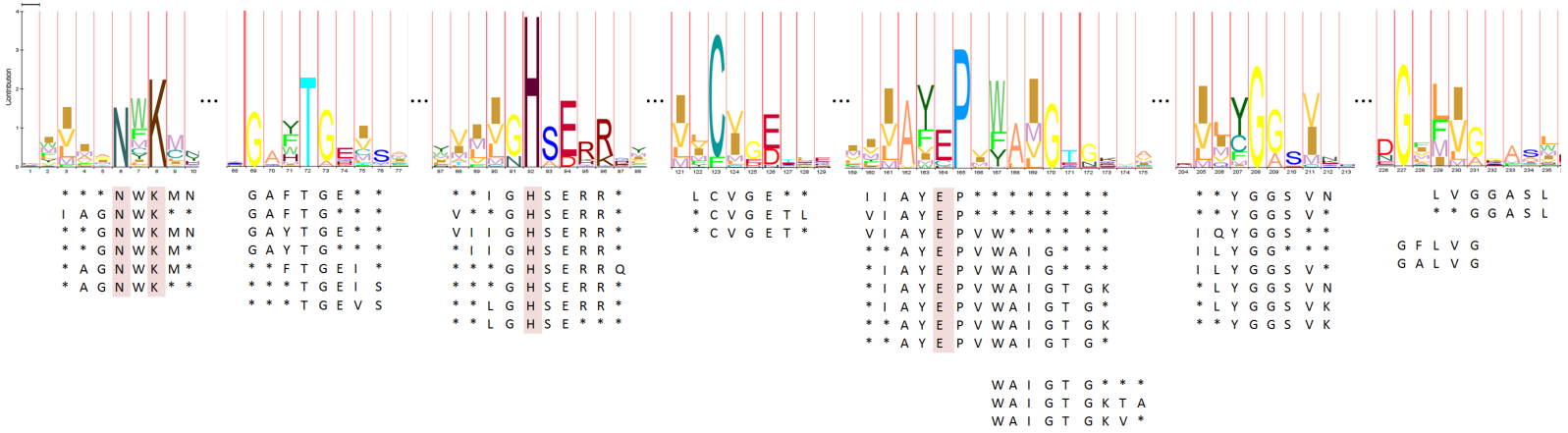


Figure 3.8: The HMM and AP Cluster comparison of TIM. The resulting seven AP Clusters of the TIM protein family cover four of the binding residues, which are highlighted in red: Asn6, Lys8, His92, and Glu164.

Our AP Synthesis Process resulted in AP Clusters with dendrograms that trace the iterative merge of our hierarchical clustering algorithm (Fig. 3.7).

Ubiquitin Discussions

Ubiquitin contains seven lysine residues (Lys6, Lys11, Lys27, Lys29, Lys33, Lys48, and Lys63) that link other ubiquitins to form a poly-ubiquitin chain[94, 223, 85](Fig. 3.9). Our

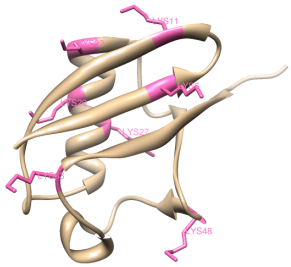


Figure 3.9: The 3D structure of ubiquitin (PDB ID: 1UBQ). It has seven binding residues (in pink).

resulting AP Clusters correspond to six of the seven binding residues as listed above. The remaining Lys33 is found in an AP Cluster with only a single distinct pattern discovered with high statistical significance; thus, this AP Cluster stands out as a significant functional group.

For ubiquitin, our AP Clusters are short alignments of patterns that agree with the emission probabilities of the HMM logo from Pfam (Fig. 3.7). The eight AP Clusters cover the seven binding residues and agreed with the Pfam HMM emission probabilities but not with PROSITE's consensus motif (Pattern PS00299), which has 198 true positives and 197 false negatives when matched against the sequences in UniProtKB/Swiss-Prot.

3.3.3 The Pfam TIM Protein Family

TIM Results

To further explore the biological significance of AP Clusters, we applied our method to the TIM protein family. The input sequences are uniquely identified in Pfam by the family identification number, PF00121, which contains 56 seed sequences that have a maximal length of 244 and was downloaded February 19th, 2013 from Pfam release 27. In this experiment, parameters were the same as cytochrome c, except for a *minimum occurrence* of 10 for the Pattern Discovery Step. The 51 discovered patterns were compressed into nine AP Clusters, which contained four binding residues in three binding segments with an Average Cluster Quality of 0.40.

TIM Discussions

The AP Cluster IAGNWKMN covers Asn6 and Lys8, which are residues that bind the DHAP or GAP substrate[106]. These substrates are initially attracted to these residues through electrostatic interactions to establish the first step of the catalytic reaction[106]. Another AP Cluster covers Thr72, which is found to interact with Lys8 and Glu94, which is covered by the IAGNWKMN, of another TIM via hydrogen bonds for dimerization [194]. These discovered AP Clusters are important because the enzyme is only active in the oligomeric [157]; this hypothesizes that there is cooperation between these residues.

The AP Cluster VIGHSERRQ covers His92 in addition to Glu94. His92 is crucial for the enzymatic reaction by cooperating with Glu164, which is covered by [IV]IAYEPVWAIGTGK.

According to the classic mechanism [130, 208], Glu164 plays the role of the general base catalyst by abstracting a proton from the pro(R) position of carbon 1 of DHAP or the C-2 proton of GAP. However, the carboxylate group of Glu164 alone does not possess the basicity to abstract a proton. Hence, Glu164 is assisted by His92, the general acid, to donate a proton to stabilize the negative charge building up on C-2 carbonyl oxygen, effectively stabilizing the planar endediol(ate) intermediate.

The AP Cluster [IV]IAYEPVWAIGTGK covers residues Tyr163, Glu164, Trp167, Gly172 and Ala175, which correspond to the structure known as loop 6 [88]that is important for the enzymatic reaction.

In addition to Glu164, Tyr163, Gly172 (the nitrogen atoms on the main chain) and Ala175 (the nitrogen atoms on the main chain) have hydrogen bonds with Trp167, Ser210 and Tyr207 respectively, in which Tyr207 and Ser210 are covered by C6 [107]

3.4 Comparisons with Existing Methods

3.4.1 Identifying Binding Residues

In the previous sections, we presented each step of our method and its capability to find binding sites and other amino acids of biological significance for cytochrome c, ubiquitin, and TIM. In this final section, we compare our AP Synthesis Process with other existing motif finding methods. Table 3.12 illustrates that our AP Clusters cover the two binding residues for cytochrome c and all seven binding residues for ubiquitin. Quantitatively, our AP Clusters cover more binding residues. Of the other methods examined, MEME

comes closest to finding all the binding residues for cytochrome c, ubiquitin, and TIM. We measure the quality of patterns by percentage coverage (C) and information entropy (H) of the pattern in the data. Due to the statistical significance of the patterns found by our Pattern Discovery Step, our entropy is lower, and thus more stable than those obtained from the compared methods; however, the sequence coverage of our high quality AP Clusters is lower than others since we take into account only the strong statistically significant patterns in the AP Clusters.

3.4.2 Strong, Weak, and Conserved AP Clusters

Therefore, qualitatively, to improve our coverage while maintaining low entropy, we added two extended steps in the AP Cluster Refinement Step to generate Weak and Conserved AP Clusters, respectively. First, the Weak AP Clusters added the highest scoring sequence matches from each of the uncovered sequences within their relative position to improve the overall coverage. Second, the Conserved AP Clusters restrict occurrences of Weak AP Clusters to have the conserved columns of the AP Cluster. Hence, the Conserved AP Cluster has a higher coverage than its corresponding AP Clusters, but a lower entropy than its Weak AP Clusters. Therefore for entropy, the Conserved AP Cluster has higher entropy than its corresponding AP Clusters, but a lower entropy than its Weak AP Clusters. The results from the different AP Clusters are presented in Table 3.13 where we measure the patterns by percentage coverage (C) and information entropy (H) of the pattern in the data.

Table 3.12: Binding Residues Results Compared with Other Methods

Method	Binding Residue	Length	C	H
Cytochrome c:				
AP Cluster	His18	11	0.35	0.58
	Met62	8	0.16	0.51
MEME [14]	His18	15	1	0.61
	Met62	29	0.25	0.41
Gibbs [109]	His18	12	1	0.46
BLOCKS [77]	His18	16	0.97	0.61
	Met62	5	0.97	0.58
CONSENSUS [80]	His18	8	1	0.51
PROJECTION [32]	His18	3	0.75	0.24
	Met62	3	0.75	0.34
Ubiquitin:				
AP Cluster	Lys6, Lys11	6	0.12	0.38
	Lys27, Lys29	5	0.08	0.11
	Lys33	5	0.12	0.00
	Lys48	14	0.37	0.53
	Lys63	7	0.12	0.21
MEME [14]	Lys48	32	0.96	0.60
	Lys11, Lys27	29	0.96	0.64
	Lys29, Lys33	7	0.03	0.13
Gibbs [109]	Lys48	12	0.65	0.28
BLOCKS [77]	Lys48	10	1	0.53
CONSENSUS [80]	Lys48	16	1	0.53
PROJECTION [32]	Lys48	4	1	0.57
	Lys27, Lys29	5	0.12	0.35
	Lys6	8	0.12	0.42
	Lys63	8	0.07	0.12
TIM:				
AP Cluster	Asn6, Lys8	5	0.52	0.37
	His92	10	0.31	0.84
	Glu164	14	0.77	0.26
MEME [14]	His92	41	0.98	0.59
	Glu164	24	0.95	0.49
Gibbs [109]	Glu164	9	1	0.24
BLOCKS [77]	Asn6, Lys8	11	0.77	0.75
	His92	41	0.77	0.65
	Glu164	12	0.77	0.47
CONSENSUS [80]	Glu164	8	1	0.24
PROJECTION [32]	Glu164	8	1	0.36
	His92	8	1	0.26

Table 3.13: Compared Refined AP Cluster

Protein Family	Binding Residue	Strong H	Strong C	Weak H	Weak C	Conserved H	Conserved C
Cyto c	His18	0.58	0.35	0.64	1	0.59	0.97
Cyto c	Met62	0.51	0.16	0.74	1	0.63	0.61
Ubiq	Lys6, Lys11	0.38	0.12	0.75	1	0.38	0.12
Ubiq	Lys27, Lys29	0.11	0.08	0.68	1	0.11	0.08
Ubiq	Lys33	0.00	0.12	0.64	1	0.00	0.12
Ubiq	Lys48	0.53	0.37	0.65	1	0.53	0.37
Ubiq	Lys63	0.21	0.12	0.78	1	0.21	0.12
TIM	Asn6, Lys8	0.37	0.52	0.52	1	0.42	0.77
TIM	His92	0.31	0.84	0.40	1	0.32	0.88
TIM	Glu164	0.26	0.77	0.37	1	0.27	0.89

Length-Comparable with Existing Methods

Qualitatively, for the most statistically significant ubiquitin motif, our Weak AP Cluster and Conserved AP Cluster have lower and thus better entropy coverage than the other methods, with the exception of Gibbs, which has poorer coverage despite the lower entropy. However, it should be noted that lower and better entropy is a trade off against higher coverage; thus, our Weak AP Cluster for ubiquitin is able to discover more binding residues while maintaining comparable coverage and entropy (Figure 3.10). When Table 3.13, we note that while our AP Synthesis Process can discover all the binding residues, our AP Cluster Refinement Step allows a balance between the entropy and the coverage to achieve superior results. Overall, the entropy of our AP Clusters are superior to its motif finding contemporaries.

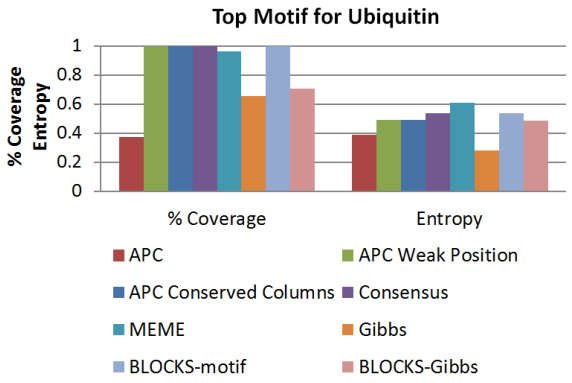


Figure 3.10: Detailed comparison of only the binding residue in the top ranked motif of the ubiquitin protein family against all the the other methods by percentage coverage and entropy. Notice the better results have higher percentage coverage that is close to 100% and lower entropy that is closer to 0%.

3.5 Chapter Conclusion

This study presents an AP Synthesis Process that brings similar patterns together, aligns and clusters them into AP Clusters with patterns as rows and aligned sites as columns to reveal both the statistically significant associations of amino acids in the protein segments as well as the conservations and variations of the amino acids of the aligned sites. The AP Clusters obtained for the cytochrome c, the ubiquitin, and the TIM protein families correspond to the protein binding segments respectively. The results of our AP Synthesis Process agree with the Pfam emission probability and render higher quality binding sites than its contemporaries. To generate high-quality AP Clusters, we found that using (1) Global Alignment as the MERGE Algorithm with (2) Hamming Distance as the SIMILARITY Score and setting (3) the TERMINATION Condition to optimal threshold yields the optimal Average Cluster Quality.

In conclusion, AP Clusters can be used to reveal functional domains across different protein families without relying on prior knowledge or clues about the consensus regions as evidenced by the experimental results obtained from the cytochrome c, ubiquitin and TIM protein families. In fact, classification results have been obtained from our method and is addressed in Section II, which is also Chapter 5. In all these cases, our AP Synthesis Process can discover all the binding residues and our Refinement Step allows a balance between the entropy and the coverage to achieve superior results. Overall, the quality of our AP Clusters is superior to its motif finding contemporaries.

As a natural extension of the presented methods, we are (1) using aligned column variations as amino acid characteristics to classify proteins [114] in Chapter 5 and (2) extending the algorithm to discover long-distance associations between AP Clusters [111] in Chapter 6. In more general cases of protein analysis, the location and the nature of the protein functional domains are not clear. The capability to overcome such difficulties marks the uniqueness and novelty of our AP Synthesis Process.

Chapter 4

Aligned Pattern Hypergraph and Hyperedge

4.1 Chapter Introduction

We approached protein sequence analysis from a data mining or pattern discovery perspective. Hence in Chapter 3, we began by identifying a set of statistically significant sequence patterns and developed an Aligned Pattern (AP) Synthesis Process by aligning and clustering similar patterns into a reduced set of Aligned Pattern Clusters (AP Clusters) for representing the similar sequence patterns in regions that might be associated with binding segments. Then we converted each AP Cluster into a synthesized probabilistic structural pattern in the form of an AP Hypergraph, the main goal and definition of this chapter. The AP Cluster aligns a set of similar patterns, whereas, like network flow, the AP Hy-

pergraph represents the flow of patterns as well as the similarity and the differences in the aligned columns as vertices. This reduced set of Aligned Pattern Directed Hypergraphs captures both the statistically significant sequence association of amino acids as well as their conservations and variations on each of the amino acids in the regions.

We then examine whether or not the AP Hypergraphs correspond to the binding segment and binding residues that reflect a protein's functionality. The three ranking criteria presented are coverage, quality, and standard residual. When our AP Synthesis Process was applied to the cytochrome c and ubiquitin protein families, we discovered a reduced set of AP Hypergraphs solutions, which corresponds to the functional binding segments and binding residues of both families . Our AP Synthesis Process obtained a set of solutions smaller when compared to the combinatorial methods, rendering a more compact yet knowledge-rich representation in the form of an AP Hypergraph than the probabilistic method. Having a smaller set with richer representation is crucial in identifying drug targets for drug discovery.

4.2 Methods

4.2.1 The Digraph Construction Step

An AP Hypergraph is a dual representation of an AP Cluster. In the Graph Construction Step, the aligned columns of an AP Cluster are grouped by matching the patterns' amino acids into a column hyperedge. The set of similar patterns, which are unaligned, is grouped and aligned into aligned patterns that form an AP Cluster first which is then converted

into an AP Hypergraph. We first introduce several related definitions. A digraph is a network of vertices and edges; the vertices are transfer points in the network flow graph, and the directed edges provide a one-way flow of the resources. In an AP Hypergraph, these resources are the set of patterns that were grouped and aligned in the AP Cluster. Thus, the full subset of patterns, $\mathbb{P} = \{P_1, P_2, ..., P_{m-1}, P_m\}$, from the patterns are the resources flowing through vertices and edges of an AP Hypergraph, which is a richer representation. The term hypergraph is due to the vertices being grouped a) vertically by aligned columns into a column hyperedge and b) horizontally by the flow of data in the patterns into a pattern hyperedge. A hyperedge is consider a collection of items without defined associations among them; however, pattern can be considered to have consecutive path association.

Definition 9 *An Aligned Pattern Directed Hypergraph (AP Hypergraph) is a* directed graph*, $G = (\mathbb{V}, \mathbb{E})$, with a set of* vertices $\mathbb{V}$ *that are connected by a set of directed* edges*, $\mathbb{E}$. Furthermore, $\mathbb{P}^G = \{p^1, p^2, ..., p^{m-1}, p^m\}$ is the set of patterns represented by the AP Hypergraph, and each pattern is of length n. Since an AP Hypergraph is an alternative representation of an AP Cluster, the pattern set and the induced data for both representations are the same (i.e. $\mathbb{P}^G = \mathbb{P}^C$ and $\mathbb{D}^G = \mathbb{D}^C$).*

Definition 10 *Let $\nu_j(\sigma)$ be a vertex of an AP Hypergraph; the vertex represents all patterns with the same amino acid, σ, within its aligned column c_j:*

$$\nu_j(\sigma) \quad = \quad \{p^i = s_i^1 ... s_i^j ... s_i^n | s_i^j = \sigma\}. \tag{4.1}$$

where n is the length of each pattern, j is the index of the aligned column c_j, and

$\sigma \in \{\Sigma \cap *\}$. *Let* $\mathbb{P}^{\nu_j(\sigma)}$ *be the set of patterns from* $\mathbb{P}^G$ *that is restricted by the vertex* $\nu_j(\sigma)$.

Thus, the set of all existing vertices in an AP Hypergraph is $\mathbb{V} = \{\nu_j(\sigma)|j = 1, ..., n, \sigma \neq \emptyset\}$, where a vertex is not created if an amino acid does not occur at the aligned column.

The resulted AP Hypergraph is constructed by applying amino acids as vertex labels of the aligned columns (Fig. 4.1). For example, a vertex ν at aligned column c_1 with residue $\sigma = H$ is labelled with $\nu_1(H)$ having the subset of patterns $\mathbb{P}^{\nu_1(H)} = \{p^3, p^6\}$.

Definition 11 *Let an edge* $\epsilon \in \mathbb{E}$ *connect two vertices:* $\nu_j(\sigma)$ *and* $\nu_{j+1}(\sigma')$. *The edge is labelled by* $\epsilon_j(\sigma, \sigma')$, , *indicating that the aligned column* c_j *has amino acid,* σ, *and the aligned column* c_{j+1} *has amino acid,* σ', *in this set of patterns,* $\mathbb{P}^G$,

$$\epsilon_j(\sigma, \sigma') \tag{4.2}$$

$$= \{p^i = s_i^1 ... s_i^j s_i^{j+1} ... s_i^n | s_i^j = \sigma, s_i^{j+1} = \sigma'\}, \tag{4.3}$$

$$\tag{4.4}$$

where j *is the index of the aligned column* c_j *and* $\sigma, \sigma' \in \Sigma(c_j) \in \Sigma(c_j) \in \{\Sigma \cap *\}$.

Again, let $\mathbb{P}^{\epsilon_j(\sigma,\sigma')} \subseteq \mathbb{P}$, *be the set of patterns that is restricted by the edge.*

For example, the edge $\epsilon_1(H, E)$ connects the vertex $\nu_1(H)$ to the vertex $\nu_2(E)$, and the edge is the subset containing the pattern $\{p^3\}$ (Fig. 4.1).

The Graph Construction Algorithm To construct the AP Hypergraph from its AP Cluster, iterate through each aligned column of the AP Hypergraph and isolate the unique

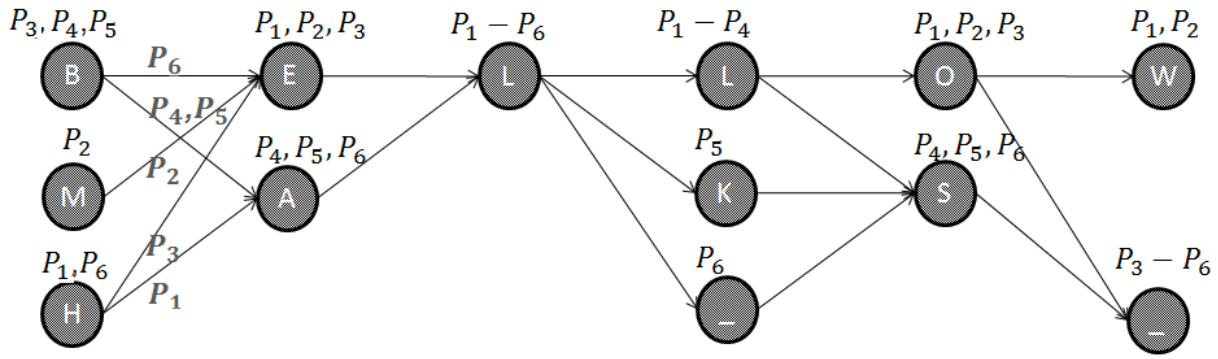


Figure 4.1: In the text example, an AP Hypergraph is converted from its AP Cluster.

amino acid of the AP Cluster to construct a table of amino acid distributions (Table 4.1). In this table, each cell represents the resulting vertex $\nu_j(\sigma)$ and stores its associated list of patterns. Furthermore, each pattern in the list references its exact occurrences in the suffix tree that are used to compute the measures. The resulting AP Hypergraph is constructed by applying the vertex labels to the AP Cluster (Fig. 4.1).

Algorithm 2 The AP Hypergraph Construction Algorithm

Require: $C = \{c_1, ..., c_n\}$, where n is the length of the AP Hypergraph
Ensure: $G = \mathbb{V} = \{\nu_j(\sigma) | j = 1...n, \sigma \neq \emptyset\}$

```
for all (aligned column c_j ∈ ℂ) do
    Initialize column hyperedge, 𝕍^j, which is a list of vertices
    for all (Aligned Pattern p_i ∈ ℂ) do
        let σ = s_i^j
        if v_j(σ) does not exist in 𝕍^j, then initialize and add to 𝕍^j then
            else add p_i to v_j(σ)
        end if
    end for
end for
```

Table 4.1: Conversion Table from aligned column to AP Hypergraph Vertices

c_1		c_2		c_3		c_4		c_5		c_6	
$\nu_1(H)$	$=\{p^1, p^6\}$	$\nu_2(E)$	$=\{p^1 - p^3\}$	$\nu_3(L)$	$=\{p^1 - p^6\}$	$\nu_4(L)$	$=\{p^1 - p^4\}$	$\nu_5(O)$	$=\{p^1 - p^3\}$	$\nu_6(W)$	$=\{p^1, p^2\}$
$\nu_1(B)$	$=\{p^2, p^4, p^5\}$	$\nu_2(A)$	$=\{p^4 - p^6\}$			$\nu_4(K)$	$=\{p^5\}$	$\nu_5(S)$	$=\{p^4 - p^6\}$	$\nu_6(\text{-})$	$=\{p^3 - p^6\}$
$\nu_1(M)$	$=\{p^3\}$					$\nu_4(\text{-})$	$=\{p^6\}$				

4.2.2 Hyperedges in the AP Hypergraph

The introduction of the AP Hypergraph reveals how vertices are associated in the form of hyperedges in order to indicate the type of associations between them. Since edges connect two vertices, hyperedges connect a subset of vertices. Three types of hyperedges are defined for the AP Hypergraph: (a) pattern hyperedge, (b) column hyperedge, and (c) association hyperedge.

Pattern Hyperedge

Before defining a pattern hyperedge, we must define the sinks, the sources, and the paths of a AP Hypergraph. The sources of an AP Hypergraph are the starting vertices that begin at the first position of an aligned column c_1 with no entering edges, and the sinks are the ending vertices that end at c_n with no exiting edges. Each pattern, $P_i \in \mathbb{P}$, is a path in the AP Hypergraph that flows through the vertices and edges. This path is of length n from a source at c_1 to a sink at c_n:

$$\mathbb{V}^i = \{\nu_1(\sigma_1), \nu_2(\sigma_2), ..., \nu_n(\sigma_n) | \sigma_1\sigma_2...\sigma_n = P_i\}, \tag{4.5}$$

where i is the index of the pattern $P_i \in \mathbb{P}$ and $\sigma \in \{\Sigma \cap *\}$. For example (Fig. 4.2(a)), a pattern of length 6, 'HELLO*', is labelled by the path from source $\nu_1(H)$ to sink $\nu_6(*)$. This pattern is a path containing vertices $\{\nu_1(H), \nu_2(E), \nu_3(L), \nu_4(L), \nu_5(O), \nu_6(*)\}$.

Column Hyperedge

The set of vertices on the same aligned column is called the column hyperedge. Each vertex contains a set of patterns at that position due to its distinct amino acid and thus, has special algebraic set properties.

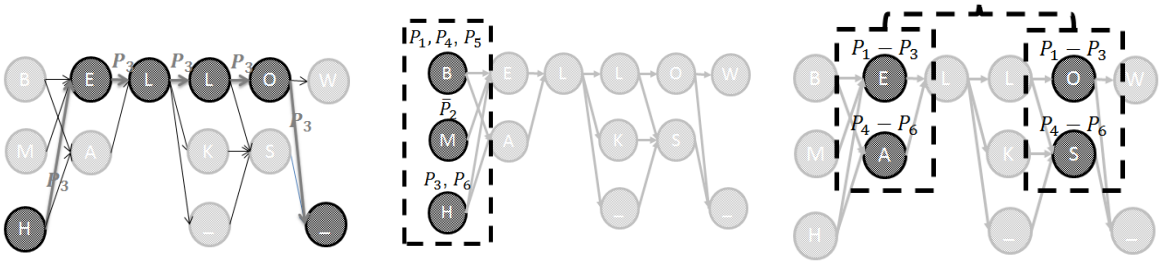
Definition 12 *Let the set of all vertices in an aligned column c_j be defined as the column hyperedge:*

$$\mathbb{V}_j \quad = \quad \{\nu_j(\sigma) | \forall \sigma \in \Sigma(c_j) \in \Sigma, \nu_j(\sigma) \neq \emptyset\}, \tag{4.6}$$

where j is the index of the aligned column c_j, and $\sigma \in \{\Sigma \bigcap *\}$.

For example (Fig. 4.2(b)), the first column hyperedge is $\mathbb{V}^1$ that contains vertices $\{\nu_1(B), \nu_1(M), \nu_1(H)\}$. In this case, the first column hyperedge contains all the sources, and the last column hyperedge contains all the sinks. The column hyperedges are ordered consecutively; they have an adjacent relationship if one is next to the other. A vertex in a column hyperedge connects only to the vertices in the column hyperedge.

The size of a column hyperedge is the number of vertices in the aligned column c_j and is also the number of distinct σ in that aligned column. The size is used to compute the entropy later and is also used to define Conserved and Variable column hyperedges. A



(a) pattern hyperedge (b) column hyperedge (c) association hyperedge

Figure 4.2: Further demonstration of the AP Hypergraph to introduce the concepts of pattern as a path and aligned column c_1 as column hyperedge: (a) pattern hyperedge, (b) column hyperedge, and (c) association hyperedge.

column hyperedge is Conserved when $|V_j| = 1$ (i.e., the column hyperedge has only one amino acid). A column hyperedge is Variable when $|V_j| > 1$ (i.e., the column hyperedge has multiple distinct amino acids values). Therefore, the Conserved column hyperedge characterises the induced data of an AP Hypergraph because it spans all the patterns and an Variable column hyperedge partitions the induced data into sub-clusters for unsupervised classification.

The Set of Patterns in a Column Hyperedge The column hyperedge, $\mathbb{V}^j$, has the following algebraic set properties defined based on its list of patterns in an AP Hypergraph(i.e. , $\mathbb{P} = \{p^1, p^2, \ldots, p^m\}$).

1. **Universal Set**: It is the entire set of patterns, $\mathbb{P}^G$, in an AP Cluster to be used to specify which patterns will be contained in a column hyperedge. The Universal set

of a column hyperedge $\mathbb{V}_j$ can be expressed as:

$$\begin{aligned}
\bigcup_{\nu_j(\sigma)\in\mathbb{V}^j} \nu_j(\sigma) &= \bigcup_{\nu_j(\sigma)\in\mathbb{V}^j} \{p^i \in \mathbb{P} | s_j^i = \sigma\} \\
&= \Big(\bigcup_{\nu_j(\sigma)\neq\emptyset} \{p^i \in \mathbb{P} | s_j^i = \sigma\} \Big) \\
&= \bigcup_{\sigma\in\Sigma(c_j)\in\Sigma} \{p^i \in \mathbb{P} | s_j^i = \sigma\} \\
&= \mathbb{P}.
\end{aligned}$$

For example for $\mathbb{V}_1$, $\nu_1(B) = \{p^1, p^4, p^5\}$, $\nu_1(M) = \{p^2\}$, and $\nu_1(H) = \{p^3, p^6\}$. These pattern subsets of each of the vertices, $\{p^1, p^4, p^5\} \bigcup \{p^2\} \bigcup \{p^3, p^6\}$, which together is the universal pattern set, $\mathbb{P}$.

2. **Disjoint Subsets**: Since the jth column hyperedge has vertices representing distinct amino acids, $\nu_j(\sigma) \cap \nu_j(\sigma') = \emptyset$ when $\sigma \neq \sigma'$, and $\sigma', \sigma \in \Sigma$. In other words, the set $\nu_j(\sigma), \forall\sigma \in \Sigma$ are pairwise disjoint.

   For example, the first aligned column that contains the vertices of the column hyperedge, $\mathbb{V}^1$, has $\{\nu_1(B), \nu_1(M), \nu_1(H)\}$. The pattern subsets $\{p^1, p^4, p^5\}$ and $\{p^2\}$ are disjoint.

3. **Empty Subset**: By construction, (j, σ) is a vertex if and only if $\nu_j(\sigma) \neq \emptyset$

   For example, $\nu_1(E) = \emptyset$ since the amino acid E does not exist at aligned column c_1.

These set properties can be generalized for the induced data of an AP Hypergraph, which is defined and further explained in the next induced data section.

Association Hyperedge

We use Mutual Information between two aligned columns to determine how much interdependency is between the data represented by the vertices of the two column hyperedges, where the induced data is used to compute all the differences. This work is being further developed under future work using cluster validity measures from Chapter 5 with principal component analysis, which is beyond the scope of this dissertation.

4.2.3 Measuring and Ranking AP Hypergraphs

The Three Measures of AP Hypergraphs

In order to rank a set of constructed AP Hypergraphs, $\mathbb{G}$,three measures are computed for each AP Hypergraph, G^l. They are measures are Coverage, AP Hypergraph Quality, and Standard Residual. We develop a ranking algorithm to rank each of the AP Hypergraphs.

Algorithm 3 The AP Hypergraph Ranking Algorithm

Require: List of $G^l \in \mathbb{G}$ and its corresponding occurrences
Ensure: Measures for each G to be used for ranking the list $\mathbb{G}$
1: **for all** $(G \in \mathbb{G})$ **do**
2: Get exact data occurrences of G
3: Compute the Measure
4: **end for**
5: Sort $\mathbb{G}$ by the Measure

Coverage The coverage accounts for the fraction of the total input sequences that are covered by the AP Hypergraph, G, in the input sequences, $\mathbb{D}^G = \mathbb{D}$, of the data space.

AP Hypergraph Quality The AP Hypergraph Quality, Q , is the average column entropy subtracted from one, where entropy is computed from the set of Aligned Patterns, $\mathbb{P}^l \in G^l$. The AP Hypergraph Quality measures the stability or reliability of an AP Hypergraph, whereas the entropy measures the randomness or variation within an AP Hypergraph. As the value of Q approaches one, the resulting AP Hypergraph is more stable. As the value of Q approaches zero, the resulting AP Hypergraph is more random. Q is expressed as:

$$Q = 1 - \frac{1}{n} \sum_{j=1}^{n} H(\mathbb{V}_j), \tag{4.7}$$

where $\mathbb{V}_j$ is the column hyperedge in the resulting AP Hypergraph.

$$H(\mathbb{V}_j) = - \sum_{\forall \sigma \in c_j} Pr(\nu_j(\sigma)) \log Pr(\nu_j(\sigma)), \tag{4.8}$$

$$Pr(\nu_j(\sigma)) = \frac{1}{m} \sum_{i=1}^{m} 1(s_i^j = \sigma) \tag{4.9}$$

where $\sigma \in \Sigma \cup \{-\} \cup \{*\}$ is the amino acid s_i^j of p_i at $\mathbb{V}_j$, and the probability $Pr(\nu_j(\sigma))$ is computed from counting the subset of patterns in $\mathbb{P}$.

Standard Residual The Standard Residual measures the statistical significance of the AP Hypergraph by comparing the actual number of occurrences, o, of all the patterns in the AP Hypergraph, against the expected number of occurrences, e, which is computed from the probability of the amino acid in the defaulted model corresponding to the AP

Hypergraph. It is written as

$$\texttt{StandardResidual} = \frac{o-e}{\sqrt{e}}, \tag{4.10}$$

where o is the actual number of occurrences of the pattern in $\mathbb{P}$ counted from the input data, $\mathbb{D}$ and e is the expected number of occurrences computed from the probability of the amino acid in defaulted random model of the AP Hypergraph as shown below:

$$\begin{aligned} e &= E[G], && (4.11)\\ &= N\Big(Pr(G)\Big), && (4.12)\\ &= N\Big(Pr(\mathbb{V}_1)Pr(\mathbb{V}_2)\ldots Pr(\mathbb{V}_n)\Big), && (4.13)\\ &= N\Big(\prod_{j=1}^{n} Pr(\mathbb{V}_j)\Big), && (4.14) \end{aligned}$$

where N is the length of the input sequence and $\mathbb{V}_j$ is the column hyperedge. To compute the probability, $Pr(\mathbb{V}_j)$, of each of the column hyperedges, $\mathbb{V}_j$, each of the vertices in the column hyperedge must be summed.

$$\begin{aligned} Pr(\mathbb{V}_j) &= Pr(\nu_j(\sigma_1)) + Pr(\nu_j(\sigma_2)) + \ldots, && (4.15)\\ &= \sum_{\forall \sigma_k \in \mathbb{V}_j} Pr(\nu_j(\sigma_k)), && (4.16) \end{aligned}$$

where the vertex $\nu_j(\sigma_k)$ exists only if σ_k is assumed to exist for that column hyperedge, $\mathbb{V}_j$, and $Pr(\sigma_k)$ is assumed to be drawn from equal probabilities such that $Pr(\sigma_k) = \frac{1}{20}$. Therefore, the final expectation is

$$e = E[G], \tag{4.17}$$

$$= N\left(\prod_{i=1}^{n}\left(\sum_{\forall\sigma_k\in\mathbb{V}_j} Pr(\nu_j(\sigma_k))\right)\right). \tag{4.18}$$

4.3 *In Silico* Biological Experiments

We conducted a biological experiment on the cytochrome c and the ubiquitin protein families to examine how the resulting AP Hypergraphs are related to the binding sites that associate with the most important functionality of the protein. There are three aspects we explored: the reduction of the set of candidate solutions from the discovered patterns to the AP Hypergraphs obtained; how each pattern in the AP Hypergraph surrounding the binding site represents a binding segment in a single strand of protein; and how binding residues correlate to their column hyperedges. Finally, we display our results underneath the pFam multiple sequence alignment to compare the differences in the representations. In the comparison, we demonstrate the overall hierarchical clustering performance of our AP Synthesis Process as well as the quality of the resulting AP Hypergraphs.

4.3.1 The UniProt Cytochrome C Protein Family

Cytochrome C Results

First, we demonstrated that by grouping similar patterns together, the AP Hypergraph reduces the number of candidate solutions to be examined without losing information. Next,

we showed that in the binding AP Hypergraphs, each pattern represents a binding segment in the protein sequence and each of the two binding sites is represented by a specific column hyperedges. The 317 sequences from the cytochrome c protein family were obtained on September 17th, 2012 from Uniprot by searching the following terms: cytochrome c; AND reviewed:yes; AND name:c*; AND mnemonic:c*; AND (name:cytochrome AND name:c); NOT name:type; NOT name:VPR; NOT name:biogenesis; NOT name:*ase; NOT (name:cytochrome AND name:b*); NOT like; NOT proba*; AND fragment:no; AND active:yes. These selected parameters should help to yield a reasonable number of input sequences for the AP Synthesis Process. For these 317 input sequences, the Pattern Discovery Step was executed with the *minimal order* of 5, which is dependent on the number of input sequences, the *minimum occurrence* of 20, and the *delta* of 0.9. The Pattern Discovery Step discovered 154 patterns from the cytochrome c protein family, where 28 patterns, or 18.18% of the total patterns, contain the proximal binding site, His18, and 23 patterns, or 14.94% of the total patterns, contain the distal binding site, Met62, resulting in a combined total of 33.12% of the discovered patterns that contain one of the two binding sites. Therefore, the set of patterns redundantly covers the two binding sites. This observation indicates that each individual pattern alone covers only a small fraction of the input sequences in the data space; therefore, a single pattern by itself cannot fully represent the rich variations of all the input sequences within the entire protein family. Hence, the AP Hypergraph, which contains a set of similar patterns that have been grouped and aligned to allow variations, provides a reduced and much richer representation of the binding segments and binding residues.

We showed that our AP Synthesis Process reduced the number of candidate solutions

without losing any information and richly captured the binding sites in the compact AP Hypergraphs where the binding segments are the patterns therein and the binding sites are the conserved columns. We ensure that all the patterns discovered are strongly statistically significant by starting with a tighter configuration to ensure the quality of the result. From this list of 154 statistically significant and non-redundant patterns obtained from the previous Pattern Discovery Step, the AP Clustering Step was executed with the following settings: the MERGE *Algorithm* as Global Alignment, the SIMILARITY *Score* as Hamming Distance, and the three TERMINATION *Conditions* include a termination score less than 0.8, a heuristics column distribution score that is greater than 0.8, and a minimum of three overlapping column matches. Then the resulting 36 AP Hypergraphs, which have an averageAP Hypergraph Quality of 0.65 (Table 4.2), were converted into the corresponding AP Hypergraph.

We found the following two results (Table 4.3): five of the AP Hypergraphs (13.89% of the total number of AP Hypergraphs) discovered contain the proximal binding site, His18; and five of the AP Hypergraphs (13.89% of the total number of AP Hypergraphs) contain the distal binding site, Met62; the combined total is 27.78%. This observation indicates that, while retaining the full information, the 154 patterns were reduced to 36 AP Hypergraphs, a total reduction of 76.62% for documentation and visualization.

As can be seen in Table 4.4, the top four resulting AP Hypergraphs correspond to the proximal and distal binding segments of the cytochrome c protein family. More specifically, 26 proximal patterns were reduced to the two top AP Hypergraphs (a 92.31% reduction) and 16 distal patterns were reduced to the two top AP Hypergraphs (a 87.50% reduction), for a combined reduction of 88.10% for these top four AP Hypergraphs.

Table 4.2: The 36 AP Hypergraphs of the Cytochrome C Family Ranked by Standard Residual (where m=the number of patterns in the AP Hypergraph, and n=length of the AP Hypergraph))

	AP Hypergraph (as regular expressions)	m	n	Quality	Coverage	Standard Residual	Binding Site
1	WGEDTLMEYLENPKKYIPGTK**M**IFAGIKKK	8	30	0.57	81	5.92E+16	Met62
2	MGDVEKGKKIFVQ[KR]CAQC**H**TVEKGGKHKTGPNL	19	33	0.43	119	5.04E+16	His18
3	QC**H**TVEKGGKHKTGPNLHGLFGRKTGQA	7	28	0.41	46	8.32E+14	His18
4	TLYDYLLNPKKYIPGTK**M**[VA]FPGLKKPQ	8	27	0.44	116	1.91E+14	Met62
5	GAGHK[QVT]GPNL[NH]GLFGRQSGTT	13	21	0.4	125	3.53E+10	
6	GFSYTDANKNKGITWGE	8	17	0.41	66	6.33E+08	
7	GEKIFKTKCAQC**H**TV	3	15	0.57	24	6.45E+07	His18
8	MGDVEKGKKIFVQKC	7	15	0.4	53	5.04E+07	
9	GPNLHGLFGRKTGQA	4	15	0.43	46	4.37E+07	
10	ERADLIAYLK[KE]ATNE	9	15	0.4	91	3.53E+07	
11	HGLFGRKTGQAPGF	9	14	0.46	70	2.10E+07	
12	IPGTK**M**AFGGLKK	4	13	0.42	136	9.06E+06	Met62
13	AANKNKGITWGE	4	12	0.5	54	1.60E+06	
14	LHGLFGR[QK]SGTT	6	12	0.42	88	1.07E+06	
15	AGYSYSAANKN	5	11	0.43	30	1.40E+05	
16	TLYDYLLNP	2	9	0.56	29	2.69E+04	
17	GQAPGFSY	2	8	0.5	27	5.57E+03	
18	TK**M**VFAG	2	7	0.57	52	3.38E+03	Met62
19	GGKHKTG	2	7	0.43	64	2.94E+03	
20	EKGKKIF	2	7	0.43	62	2.85E+03	
21	FAGLKKP	3	7	0.48	57	2.62E+03	
22	WGGGKIY	2	7	0.71	27	2.48E+03	
23	FAGIKKK	2	7	0.43	51	2.34E+03	
24	YLKKAT	1	6	1	29	1.19E+03	
25	WGEDTL	1	6	1	25	1.02E+03	
26	NCAAC**H**	2	6	0.83	30	8.68E+02	His18
27	KGAGHK	2	6	0.83	26	7.52E+02	
28	KGITW	1	5	1	49	4.46E+02	
29	GFSYT	1	5	1	42	3.83E+02	
30	FVQKC	1	5	1	39	3.55E+02	
31	DANKN	1	5	1	34	3.10E+02	
32	GYSYT	1	5	1	28	2.55E+02	
33	A**M**PAF	1	5	1	24	2.19E+02	Met62
34	C**H**AGG	1	5	1	22	2.00E+02	His18
35	FKTRC	1	5	1	20	1.82E+02	
36	LFEYL	1	5	1	20	1.82E+02	

Table 4.3: Comparing the Number of AP Hypergraphs and Patternss

	Patterns Count	%overall	AP Hypergraphs Count	%overall	%Reduction
His18	28	18.18%	5	13.89%	82.14%
Met62	23	14.94%	5	13.89%	78.26%
Total	154	33.12%	36	27.78%	76.62%

Table 4.4: Comparing the Top Four AP Hypergraphs and their Patterns

	Pattern Count	AP Hypergraphs Count	%Reduction
His18	26	2	92.31%
Met62	16	2	87.50%
Total	42	4	88.10%

Cytochrome C Discussion

Our results show that the set of AP Hypergraphs discovered by our AP Synthesis Process that contain the protein binding sites – the main biological function of the protein.

In fact, the four top resulting AP Hypergraphs precisely correspond to these crucial binding segments that contain conserved columns corresponding to the binding residues.

The ten AP Hypergraphs that correlate to the two binding sites were first clustered based on their horizontal patterns in their rows and are then aligned into their column hyperedges that reveal their vertical stability. First, each AP Hypergraph contains a set of statistically significant patterns that are similar to one another. Although these patterns suggest their horizontal significance in the protein family, individually they do not identify the significance of the amino acid's conservation and variation. Thus, the stability of the column hyperedges is important for identifying the binding residue. Second, the column hyperedges of each binding AP Hypergraph show the column hyperedges in the cluster contain the conserved residue, which otherwise is not easily seen in the individual non-variable patterns. For example, consider the top two AP Hypergraphs that correspond to each of the proximal (Table 4.5and Fig. 4.3) and distal binding segments (Table 4.6and Fig. 4.4). In these Tables, the columns in bold are the conserved columns with R1= 1.0,

where R1 reflects the specificity of the residue of the site in the AP Hypergraph. The column hyperedges corresponding to the binding sites of the AP Hypergraphs have an R1 value of 1.0, that is, the amino acid for that column hyperedge is conserved in the data space. To give a precise example, consider the proximal AP Hypergraph that is ranked second. This AP Hypergraph has three conserved columns with an R1 value of 1.0: Gln16, Cys17, and His18. The His18 conserved column is the proximal binding residue, and the Cys17 binds an adjacent corner on the heme ligand. Similarly, the conserved column representing Met62 in the distal AP Hypergraph acts as the distal binding residue. The other conserved columns can be used to identify other important functions in the protein.

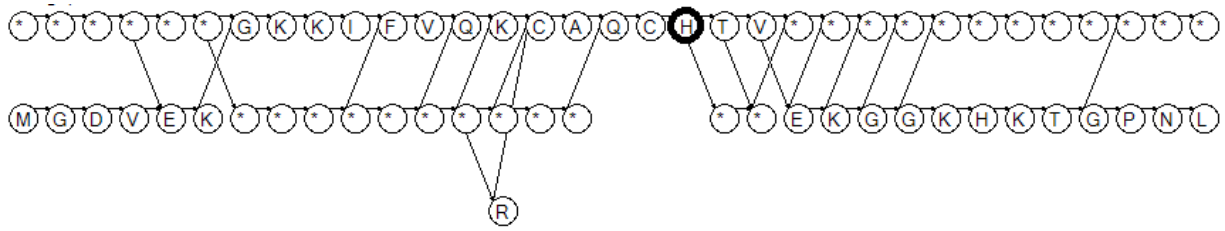


Figure 4.3: The corresponding proximal AP Hypergraph of the cytochrome c family that is ranked second.

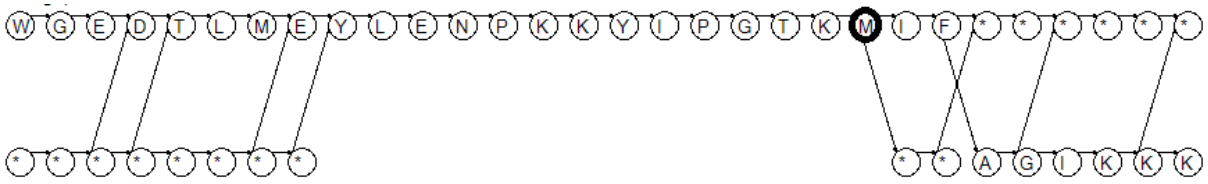


Figure 4.4: The corresponding distal AP Hypergraph of the cytochrome c family that is ranked first.

Table 4.5: The Proximal AP Hypergraph of the Cytochrome C Family

patterns	Count	Position
******GKKIFVQKCAQCHTV*********	23	6.27E+04
****EKGKKIFVQKCAQCHT**********	23	1.32E+04
MGDVEKGKKIFVQKCAQCHTVEKGGKHKTG	20	7.50E+07
******GKKIFVQKCAQCHTVEKGGKHKTG	20	1.16E+06
**********KCAQCH**********	57	1.59E+01
**********CAQCH**********	89	2.58E+03
**********RCAQCHT**********	21	1.38E+01
**********CAQCHT**********	76	3.01E+01
**********FVQKCAQCHTVE********	27	5.88E+02
**********QKCAQCHT**********	32	6.38E+01
**********QKCAQCHTVEKGGKHKTG	23	6.33E+04
**********KCAQCHTVEKG******	30	4.91E+01
**********KCAQCHTV*********	51	1.73E+01
**********CAQCHTV*********	65	3.10E+01
**********CAQCHTVEK*******	34	1.30E+01
**********CAQCHTVE********	49	2.41E+01
**********QCHTV*********	95	2.33E+03
**********QCHTVEKGG*****	45	1.75E+01
**********QCHTVE********	77	3.15E+01

Table 4.6: The Distal AP Hypergraph of the Cytochrome C Family

patterns	Count	Score
WGEDTLMEYLENPKKYIPGTKMIF******	22	1.94E+03
DTLMEYLENPKKYIPGTKM*****	26	1.30E+03
*******EYLENPKKYIPGTKMIFAGIKK*	35	2.54E+02
****TLMEYLENPKKYIPGTKMIFAGIKKK	29	7.34E+02
****TLMEYLENPKKYIPGTKMIFAG****	34	4.81E+01
********YLENPKKYIPGTKM********	81	6.51E+02
*******EYLENPKKYIPGTKMIFAG****	42	5.44E+01
*******EYLENPKKYIPGTKM********	65	2.88E+01

By matching the individual AP Hypergraphs up to the independent HMM alignment of pFam (Fig. 4.5 and Fig. 4.6), we confirmed the validity of our set of 36 AP Hypergraphs.

In addition, our proximal AP Hypergraph for cytochrome c is consistent with the proximal binding motif: [C]-x(2)-[CH], from PROSITE (PDOC00169) [16, 172] and a strong emission probability in pFam (PF00034) [175]. Moreover, our method strongly identified the distal binding in our AP Hypergraphs where PROSITE does not annotate the binding site and pFam identifies only a weak emission probability.

In conclusion, the AP Hypergraph can represent protein functions such as the binding segments and binding residues and presents a reduced set of candidate solutions and specifies their location in the protein family. In cytochrome c, the prevention of binding can block cancer progression, which is an important drug discovery for cancer treatment.

4.3.2 The UniProt Ubiquitin Protein Family

Ubiquitin Results

To study the general iterative steps and to show the overall resulting quality of AP Hypergraphs, we further applied our method to the ubiquitin protein family. The 70 sequences from the ubiquitin protein family used in our experiment were obtained on August 9th, 2012 from Uniprot by searching the following terms: name:ubiquitin; NOT name:*ase; NOT name:like; NOT name:ribosomal; NOT name:modifier; NOT name:factor; NOT name:protein; NOT name:conjugating; NOT name:activating; NOT name:enzyme; AND reviewed:yes; AND mnemonic:UB*. These adopted parameters help to yield a reasonable number of input sequences for our study. For these 70 input sequences, the Pattern Discovery Step was executed with the *minimal order* of 10 due to the number of input sequences, the *minimum occurrence* of 20, and the *delta* of 0.9 to yield a proper size of

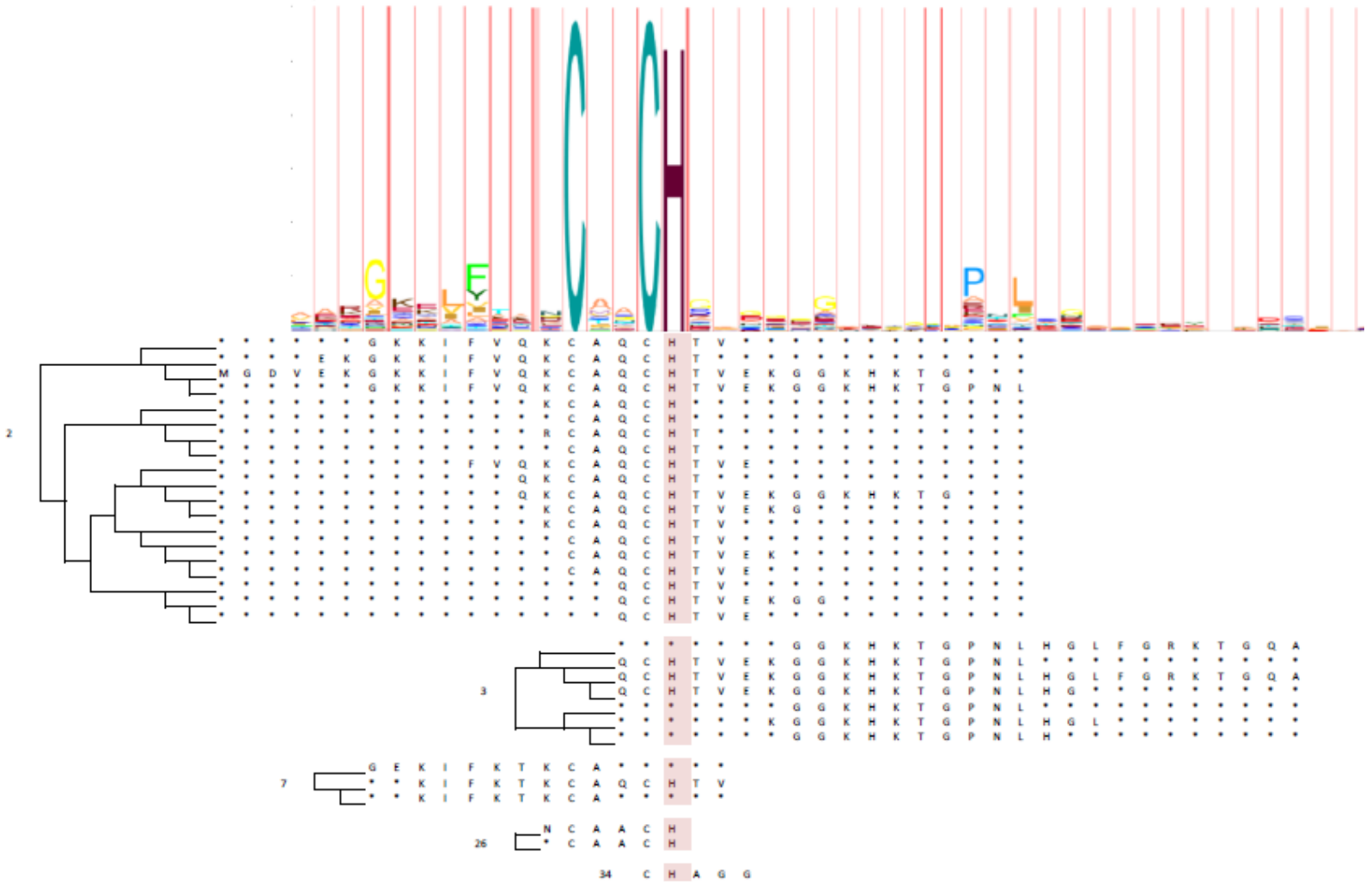


Figure 4.5: The His 18 proximal binding residues of the cytochrome c protein family is conserved column with R1= 1.0.

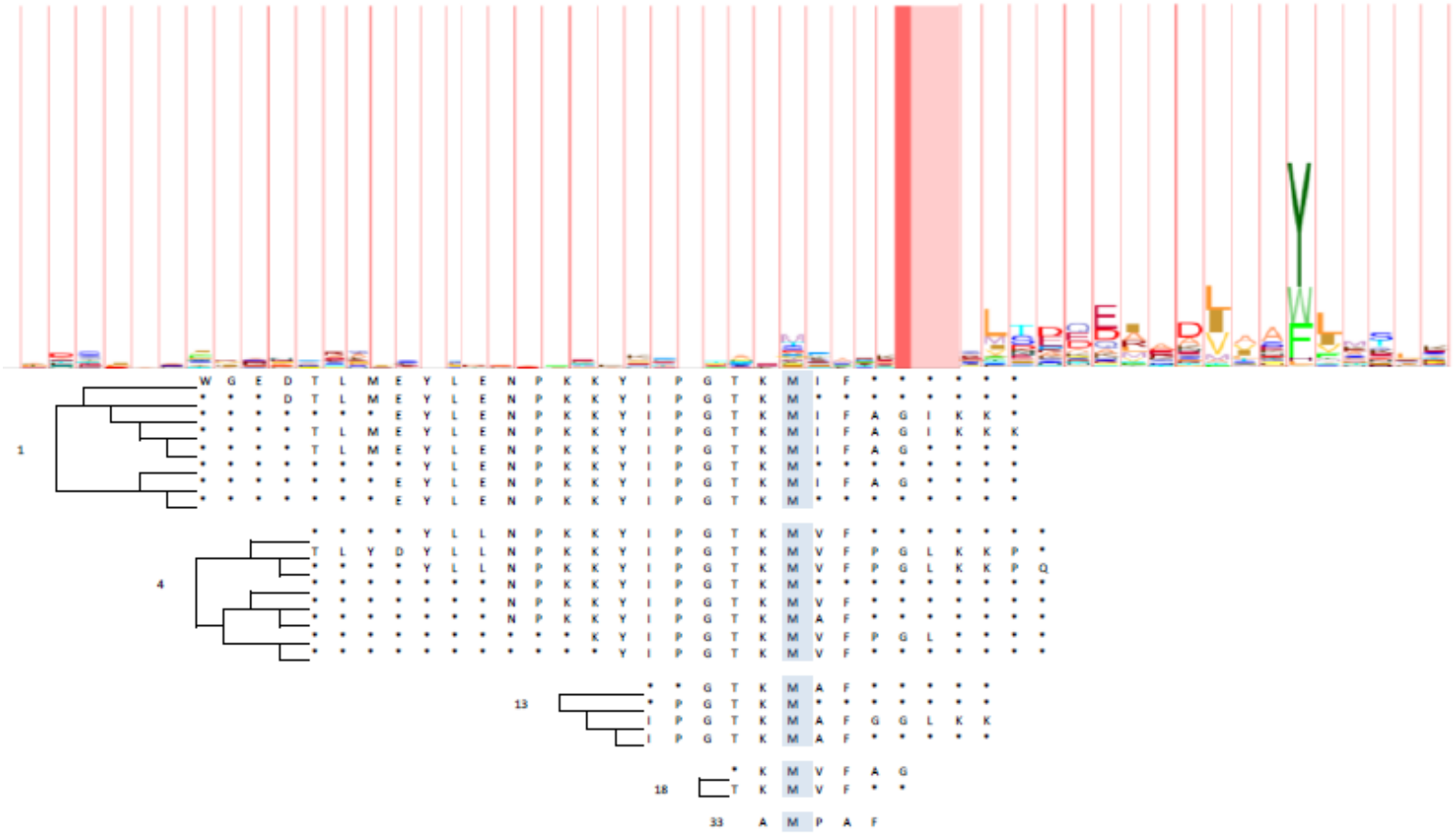


Figure 4.6: The Met 62 proximal binding residues of the cytochrome c protein family is conserved column with R1= 1.0.

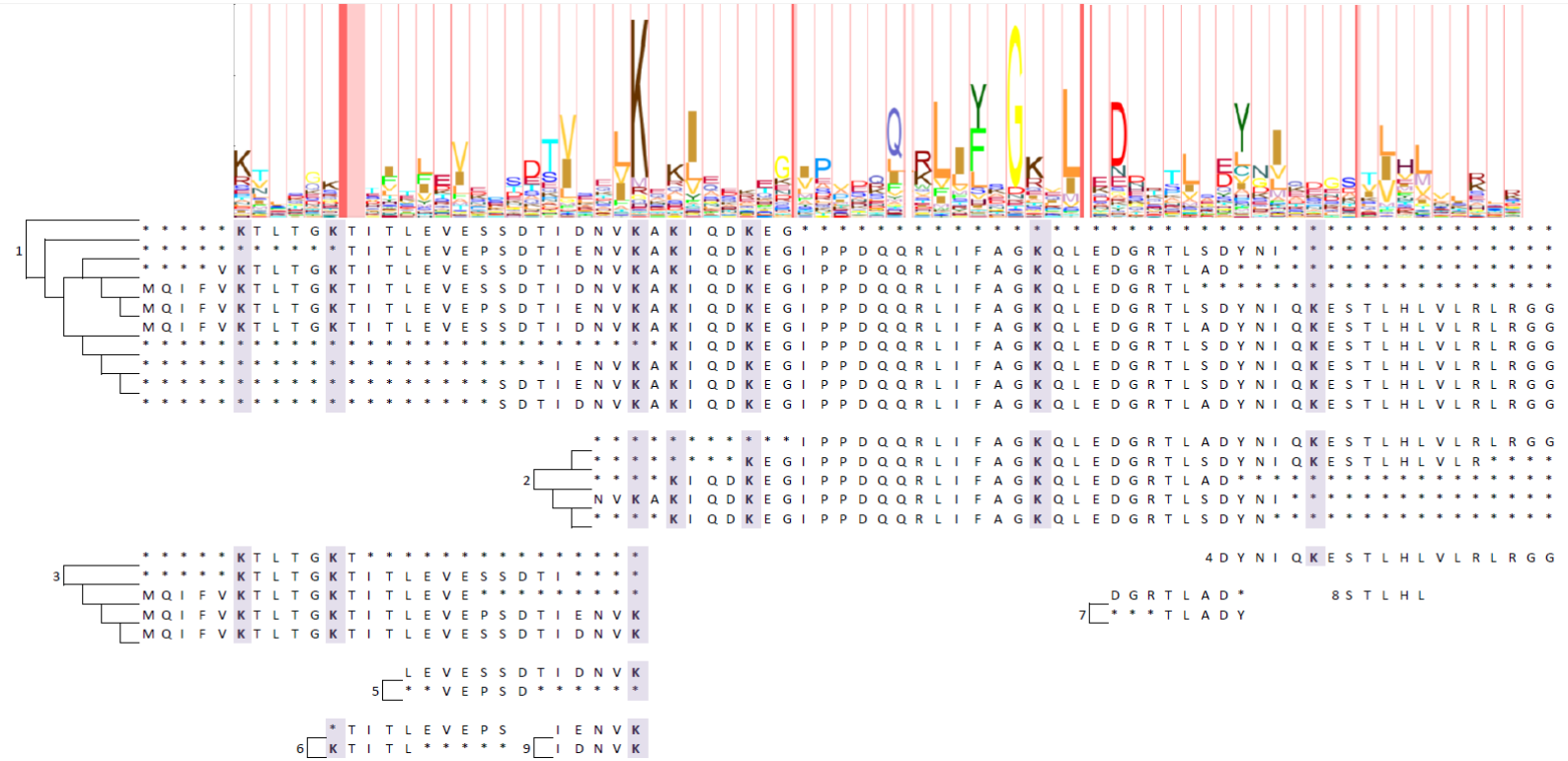


Figure 4.7: The seven Lys binding residues of the ubiquitin protein family are highlighted in the AP Hypergraph: Lys6, Lys11, Lys27, Lys29, Lys33, Lys48, and Lys63. Six of the seven binding sites are discovered, all except Lys29, are conserved column with R1= 1.0.

the results for the study. Table 4.7 shows the thirty discovered patterns, where all except five of the patterns contained the seven binding residues. Nevertheless, these patterns still corresponded to the conserved amino acids around the binding residues. Therefore, all the discovered patterns indicate important functionality in the ubiquitin protein family, such as the binding site or the areas next to the binding site. Once again, each pattern on its own occurs only a few times, and has only a low frequency count for representing the binding segments of this protein family. Since protein binding segments exhibit considerable variability, AP Hypergraphs represent the protein family's functional binding sites more explicitly and effectively.

From this list of 30 statistically significant patterns obtained from the previous Pattern Discovery Step, the AP Clustering Step was executed with the the same parameters as before. We demonstrated the efficacy of our AP Synthesis Process by showing the reduced set of 9 AP Hypergraphs and their binding sites (Table 4.8). s

Ubiquitin Discussion

Our resulting AP Hypergraphs correspond to six of the seven binding sites: Lys6, Lys11, Lys27, Lys33, Lys48, and Lys63. The remaining Lys33 is found in an AP Hypergraph with only one pattern and thus stands out as a significant functional group with a distinct pattern discovered with high statistical significance in the Pattern Discovery Step. Surprisingly, one of the Lysine identified by Pfam, which is not one of the seven binding sites, is a non-conserved Arginine(R) in our AP Hypergraph. This Lysine is not a binding site, but just another amino acid in the protein.

Table 4.7: Statistically Ranked Patterns Discovered from the Sequences of the ubiquitin Family

Ranking	Pattern	Frequency	Score	Binding Residue
1	MQIFV**K**TLTG**K**TITLEVEPSDTIENV**K**A**K**I QD**K**EGIPPDQQRLIFAG**K**QLEDGRTLSDYN IQ**K**ESTLHLVLRLRGG	21	5.44E+44	Lys6, Lys11, Lys27, Lys29, Lys33, Lys48, Lys63
2	MQIFV**K**TLTG**K**TITLEVESSDTIDNV**K**A**K**I QD**K**EGIPPDQQRLIFAG**K**QLEDGRTLADYN IQ**K**ESTLHLVLRLRGG	15	2.86E+44	Lys6, Lys11, Lys27, Lys29, Lys33, Lys48, Lys63
3	SDTIENV**K**A**K**IQD**K**EGIPPDQQRLIFAG**K**Q LEDGRTLSDYNIQ**K**ESTLHLVLRLRGG	24	1.25E+33	Lys27, Lys29, Lys33, Lys48, Lys63
4	SDTIDNV**K**A**K**IQD**K**EGIPPDQQRLIFAG**K**Q LEDGRTLADYNIQ**K**ESTLHLVLRLRGG	17	7.59E+32	Lys27, Lys29, Lys33, Lys48, Lys63
5	MQIFV**K**TLTG**K**TITLEVESSDTIDNV**K**A**K**I QD**K**EGIPPDQQRLIFAG**K**QLEDGRTL	17	4.76E+31	Lys6, Lys11, Lys27, Lys29, Lys33, Lys48
6	IENV**K**A**K**IQD**K**EGIPPDQQRLIFAG**K**QL EDGRTLSDYNIQ**K**ESTLHLVLRLRGG	32	3.48E+31	Lys27, Lys29, Lys33, Lys48, Lys63
7	V**K**TLTG**K**TITLEVESSDTIDNV**K**A**K**IQD **K**EGIPPDQQRLIFAG**K**QLEDGRTLAD	17	1.59E+30	Lys6, Lys11, Lys27, Lys29, Lys33, Lys48
8	TITLEVEPSDTIENV**K**A**K**IQD**K**EGIPPD QQRLIFAG**K**QLEDGRTLSDYNI	24	8.80E+28	Lys27, Lys29, Lys33, Lys48
9	**K**IQD**K**EGIPPDQQRLIFAG**K**QLEDGRTL SDYNIQ**K**ESTLHLVLRLRGG	39	7.43E+27	Lys29, Lys33, Lys48, Lys63
10	**K**EGIPPDQQRLIFAG**K**QLEDGRTLSDY NIQ**K**ESTLHLVLR	44	3.66E+23	Lys33, Lys48, Lys63
11	IPPDQQRLIFAG**K**QLEDGRTLADYNIQ **K**ESTLHLVLRLRGG	20	3.38E+23	Lys48, Lys63
12	NV**K**A**K**IQD**K**EGIPPDQQRLIFAG**K**QLE DGRTLSDYNI	36	6.15E+21	Lys27, Lys29, Lys33, Lys48
13	**K**IQD**K**EGIPPDQQRLIFAG**K**QLEDGRT LSDYN	44	5.20E+18	Lys29, Lys33, Lys48
14	**K**IQD**K**EGIPPDQQRLIFAG**K**QLEDGRT LAD	19	2.23E+16	Lys29, Lys33, Lys48
15	**K**TLTG**K**TITLEVESSDTIDNV**K**A**K**IQD **K**EG	19	8.01E+15	Lys6, Lys11, Lys27, Lys29, Lys33
16	MQIFV**K**TLTG**K**TITLEVEPSDTIENV**K**	25	1.17E+15	Lys6, Lys11, Lys27
17	MQIFV**K**TLTG**K**TITLEVESSDTIDNV**K**	23	8.48098E+14	Lys6, Lys11, Lys27
18	DYNIQ**K**ESTLHLVLRLRGG	62	2.40964E+11	Lys63
19	MQIFV**K**TLTG**K**TITLEVE	60	17382565255	Lys6, Lys11
20	**K**TLTG**K**TITLEVESSDTI	26	1135719784	Lys6, Lys11
21	LEVESSDTIDNV**K**	26	7757459.08	Lys27
22	TITLEVEPS	28	28304.96142	
23	**K**TLTG**K**T	67	3796.714675	Lys6, Lys11
24	DGRTLAD	23	1298.702247	
25	STLHL	69	1102.599421	
26	**K**TITL	67	315.8836468	Lys11
27	IENV**K**	38	309.1891137	Lys27
28	VEPSD	28	260.0761993	
29	TLADY	23	191.1286116	
30	IDNV**K**	29	180.0682775	Lys27

Table 4.8: The 36 AP Hypergraphs of the ubiquitin Family Ranked by Standard Residual (where m=the number of patterns in the AP Hypergraph, and n=length of the AP Hypergraph))

	AP Hypergraph(as regular expressions)	m	n	Quality	Coverage	Standard Residual	Binding Site
1	MQIFVKTLTGKTITLEVE[SP]S DTI[DE]NVKAKIQDKEGIPPDQ QRLIFAGKQLEDGRTL[SA]DYN IQKESTLHLVLRLRGG	10	76	0.31	61	4.7E+39	Lys6, Lys11, Lys27, Lys29, Lys33, Lys48, Lys63
2	NVKAKIQDKEGIPPDQQRLIFAG KQLEDGRTL[SA]DYNIQKESTL HLVLRLRGG	5	52	0.5	67	3.3E+29	Lys27, Lys29, Lys33, Lys48, Lys63
3	MQIFVKTLTGKTITLEVEP[SP] DTI[ED]NVK	5	27	0.34	67	2.7E+14	Lys6, Lys11, Lys27
4	DYNIQKESTLHLVLRLRGG	1	19	1	62	2.2E+12	Lys63
5	LEVE[SP]SDTIDNVK	2	13	0.31	54	1.0E+07	Lys27
6	KTITLEVEPS	2	10	0.4	68	4.0E+05	Lys11, Lys27
7	DGRTLADY	2	8	0.5	24	1.4E+04	
8	STLHL	1	5	1	69	1.7E+03	
9	I[ED]NVK	2	5	0.8	67	1.2E+03	Lys27

For ubiquitin, our AP Hypergraphs are short alignments of patterns that agree with the emission probabilities of the pFam profile HMM (Fig. 4.7). All eight AP Hypergraphs discovered agreed with the pFam HMM emission probability. Surprisingly, our results differs from PROSITE's consensus motif (PDOC00271), which missed 172 ubiquitin proteins. In drug discovery, preventing the linking of ubiquitin to its binding proteins via its binding site inhibits cancer growth.

4.4 Chapter Conclusion

Our AP Synthesis Process greatly reduces the number of AP Hypergraphs in comparison with other methods. This is due to the fact that the AP Clustering Step starts with input patterns from the Pattern Discovery Step rather than the entire input search space. Hence, it drastically and controllably reduced the search space. From the application aspect, using

data from two Uniprot protein families (cytochrome c and ubiquitin), the majority of top-ranking AP Hypergraphs correspond to their protein binding segments. The resulting cytochrome c binding AP Hypergraphs agree with the pFam emission probability. An AP Hypergraph represents a set of patterns as the horizontal rows and its column hyperedges as the vertical columns, which can be further evaluated for amino acid conservations. In fact, for cytochrome c, the proximal and distal binding residues correspond to conserved columns with R1 of 1.0. In addition, the distal AP Hypergraph identifies one conserved column with R1 of 1.0 as the binding residue, which is not identified in PROSITE or pFam. While the ubiquitin AP Hypergraphs agree with pFam emission probability, six of the seven binding residues are successfully identified in the AP Hypergraph.

In conclusion, AP Hypergraphs can be used to reveal functional domains across different protein families without relying on prior knowledge or clues about the consensus regions. Currently, we are using column hyperedge variations as amino acid characteristics to classify protein species and gene labels. We are also extending the algorithm to discover interdependencies within AP Hypergraphs and long-distance associations among AP Hypergraphs. In more general cases of protein analysis, the function and the nature of the protein function are not clear; thus, the capability that overcomes such difficulties marks the uniqueness and novelty of our AP Synthesis Process. In the broader sense, this knowledge is essential for understanding the proteins involved in epigenetics for drug discovery. The development of cancer generally increases with age and with the ageing baby-boomer population. It is crucial for drug companies to finding cost-saving and time-saving techniques for drug discovery.

Chapter 5

Cluster Validity Measures

5.1 Chapter Introduction

As biosequences expand quickly and increases in size and complexity, it is difficult to conduct effective analysis on the full protein sequences. Thus, identifying conserved sequence patterns is considered to be important for studying essential disjoint and joint functions in a protein. It is well recognized that conserved protein patterns are functionally essential because they are evolutionarily conserved [137] whereas mutated amino acids in these conserved regions may reflect special functionality that has evolutionarily diverged into sub-classes [51]. It is hoped that the conserved patterns and mutated amino acids may shed light on the protein class characteristics. In the past, biological ground truths were incorporated as class labels (such as protein family, gene function, species taxonomy) for each protein sequence so as to reveal inherent class characteristics.

In response to the above observations and challenges, the second important contribution of this dissertation is to use simple cluster validity measures to reveal relevant information from Aligned Pattern Clusters (AP Clusters) [115]. More specifically, the cluster validity measures reveal how distinctive entities of representation, such as Aligned Patterns, AP Clusters, distinct amino acids, and column of aligned amino acids, reveal class characteristics of the protein regions that is due to its inherent functionalities. Since AP Clusters contain aligned statistically significant patterns with strongly associated correlations, their class characteristics can be effectively revealed by the variation of amino acids on their sites as they are parts of the association between the class functional patterns. To evaluate the conservations and variations that reveal similarities and differences between sample segments and sites in AP Clusters, several cluster validity measures are proposed. It is found that because of the compact functional information being brought into the AP Clusters, the measures proposed in this dissertation are very effective at revealing class characteristics as confirmed by the ground truth and unaffected by external biases from collected class labels. In our later experiments, we do find that AP Clusters correspond to gene classes better than taxonomical classes.

To evaluate how the proposed cluster validity measures could reveal the inherent class characteristics objectively, a synthetic dataset comparing the various degrees of mislabelling was analyzed. *In silico* biological case studies on SSAT and cytochrome c were conducted to showcase the external and internal measures. Furthermore, large scale experiments on two protein families with two different class labellings were used to study the relationships between the internal and external measures with correct, ambiguous, or even incorrect class labels. Finally, we compared the results of our internal and external measures to

those results from SVM and HMM classification algorithms. Experimental results show that our clustering algorithm using external and internal cluster validity measures discovers essential amino acid variations and their relation with inherent class characteristics without requiring training. Thus, training biases, which are common to classification algorithms are avoided.

With great potential in revealing group variations, our cluster validity measures can help biochemists concentrate on identifying specific amino acids and sites with variations that are most likely to correspond to functionality or class characteristics.

This dissertation chapter is organized as follows: the methodology section describes the proposed methodology; the results section provides the results and discussion to *In Silico* and synthetic artifical experiments as evidence of the effectiveness of the proposed algorithm; and the conclusion seciton contains the concluding remarks.

5.2 Methodology

The unsupervised algorithm first discovers and clusters sequence patterns into AP Cluster using our previously developed algorithm from Chapter 3 to discover patterns and cluster align pattern [115]. Then, cluster validity measures (Table 5.3) are incorporated to measure the association of the AP Cluster and their representations with class characteristics of the protein family.

The purpose and use of cluster validity measures are to show how an AP Cluster and its representations could reveal its inherent class characteristics. Here, a text example

(Table 5.1) is examined in detail and presented for clearer understanding of the cluster validity measures. The class labels adopted here have no functional meaning as those related biological classes; they are just class names. These dataset (Table 5.1) contains three functional patterns of the English words, HELLO, MELLOW, and BELLOW, which are embedded in fifteen multiple sequences, associated with three class characteristics: happy, sad, angry.

Recall that the text example (Table 5.1) presents three sequence patterns discovered in the Pattern Discovery Step. The dataset contains three functional patterns of the English words, HELLO, MELLOW, and BELLOW, which are embedded in fifteen multiple sequences $\mathbb{S} = \{s^1, ..., s^{15}\}$ (Table 5.1). The letters outside the patterns are stochastically generated from the 26 characters of the English alphabet that are identically and independently distributed.

Table 5.1: Example of Patterns $\bar{p}^1$ =HELLO, $\bar{p}^2$ =MELLOW, and $\bar{p}^3$ =BELLOW

$\mathbb{S}$	The Input Sequences	Class
s^1	bdxejrtewkwkHELLOkcmstsjavtpi	happy
s^2	nfixtHELLOuzdovcaaxnkjfjcvwk	happy
s^3	dimtndvkjmkHELLObkcmstsj	happy
s^4	tzhgarzofdHELLOpwkxmc	happy
s^5	tyjxjqnyHELLOwmopemlqfgptnwnq	happy
s^6	kntywtoaxMELLOWbtiasycma	happy
s^7	jilxchitivMELLOWriiiweyfzgvuyaa	happy
s^8	hmlzvMELLOWorgfeb	sad
s^9	xhmlzvqgcanyMELLOWgbfj	sad
s^{10}	vqgcanyffcMELLOWvcnsnjvalbdvr	angry
s^{11}	cbpyhejgkinrphceBELLOWndwzahvkitagtt	sad
s^{12}	ndwlofBELLOWscktbucwqnboeaaklknsrmur	angry
s^{13}	fzomphnlrqhupkqBELLOWyutpfu	angry
s^{14}	skwybrfiBELLOWyvxjdijwqjvs	angry
s^{15}	nknhqexqieaBELLOWybnvrhpnsjnfms	angry

5.2.1 Cluster Validity Measures to Reveal Class Characteristics

Since an AP Cluster brings rich yet compact information of a protein region, we could relate different aspects of an AP Cluster to the class characteristics of the protein. Hence we introduce different representations in or of an AP Cluster. A representation within the AP Cluster is a distinct entity (such as a horizontal pattern, a vertical aligned column, or a distinct amino acid), which stores its counts that can be associated with external class characteristics. There are two types of cluster validity measures (Table 5.3): 1) external measures, which used for validating the representations how they are as related to the known external class labels, and 2) internal measures, which are derived merely from data reflecting the inherent functional and class characteristics. Furthermore, three types of information theory computations are incorporated: Shannon's information entropy [169], the change in information entropy (i.e., information gain), and the mutual information. Thus, by the internal and external measures for the three information theory computations, there are six cluster validity measures in total as tabulated in Table 5.2: 1) three external measures, Class Entropy (H) for representations (patterns, AP Clusters, and distinct amino acids), Class Mutual Information (R2) for column hyperedges, and Class Information Gain (IG) for column hyperedges, and 2) three internal measures for column hyperedges, Entropy Redundancy (R1), Normalized Sum of Mutual Information Redundancy (SR2) and Normalized Sum of Information Gain (SIG).

Table 5.2: Summary of Cluster Validity Measures

	External Measures (using External Class Labels)	Internal Measures (using Data Alone, Summed and Normalized)
1. Entropy	External Entropy (Normalized Class Information Entropy) $H_Y(R) = -\frac{1}{log(\lvert Y \rvert)}\Big[\sum_{y_i \in Y} pr(y_i) log\Big(pr(y_i)\Big)\Big]$. For each representation (Table 5.3)	Internal Entropy (Normalized Amino Acid Information Entropy) $H(c_i) = H = -\frac{1}{log(\lvert \Sigma(c_i) \rvert)}\Big[\sum_{\sigma(c_i) \in \Sigma(c_i)} pr(\sigma(c_i)) log\Big(pr(\sigma(c_i))\Big)\Big]$. $R1 = 1 - H(c_i)$
2. Info Gain	External Information Gain (Class Information Gain) $\Delta H_Y(c_j) = \frac{1}{H_Y(c_j)}\Big[H_Y(c_j) - \sum_{\sigma(c_j) \in \Sigma(c_j)} \Big(w(\sigma(c_j)) H_Y(\sigma(c_j))\Big)\Big]$, where $w(\sigma(c_j)) = \frac{count(\sigma(c_j))}{count(c_j)}$. Recall $\{\sigma(c_j) \lvert \sigma \in p_i, \forall p_i \in \mathbb{P}\}$	Internal Information Gain (Normalized Sum of Information Gain) $IG = \Delta H(c_i \lvert c_j) = \frac{1}{H(c_i \lvert c_j)}\Big[H(c_i \lvert c_j) - \sum_{\sigma(c_j) \in \Sigma(c_j)} \Big(w(c_i \lvert \sigma(c_j)) H(c_i \lvert \sigma(c_j))\Big)\Big]$, where $w(c_i \lvert \sigma(c_j)) = \frac{count(c_i \lvert c_j)}{count(c_j)}$. $SIG(c_i) = \frac{1}{n} \sum_{j=1}^{n} IG(c_i, c_j)$.
3. Mutual Info	Mutual Information External Joint Entropy (Joint Entropy with Class) $H(c_i, Y) = -\sum_{\sigma(c_i) \in \Sigma(c_i)} \Big[\sum_{y_i \in Y} pr(\sigma(c_i), y_i) log_2\Big(pr(\sigma(c_i), y_i)\Big)\Big]$. External Mutual Information (Class Amino Acid Mutual Information) $R2(c_i, Y) = \frac{1}{H(c_i, Y)}\big[H(c_i) + H(Y) - H(c_i, Y)\big]$.	Internal Joint Entropy (Joint Entropy by Amino Acid) $H(c_i, c_j) = \sum_{\sigma(c_i) \in \Sigma(c_i)} \Big[\sum_{\sigma(c_j) \in \Sigma(c_j)} pr(\sigma(c_i), \sigma(c_j)) log_2\Big(pr(\sigma(c_i), \sigma(c_j))\Big)\Big]$. Internal Mutual Information (Normalized Sum of Mutual Information Redundancy) $R2(c_i, c_j) = \frac{1}{H(c_i, c_j)}\big[H(c_i) + H(c_j) - H(c_i, c_j)\big]$, $SR2(c_i) = \frac{1}{n} \sum_{j=1}^{n} R2(c_i, c_j)$.

External Measures with Class Labels

To evaluate the class characteristics of an AP Cluster, we would like to find out how each representation is related to class characteristics. To this end, we consider each of the following representations from an AP Cluster: the AP Cluster by itself, the column hyperedges, and the distinct amino acid in a column hyperedge. Hence, the distribution of the class labels associated with each of the representations is used to calculate the external measures, thereby measuring the association between the representation and the class labels.

To generalize the representations from patterns to column hyperedges, we first introduce the notion of class profile.

Definition 13 *The class profile of a representation is an n-tuple of ordered pairs that store the name and the count of each class. Let* $\vec{Y} = \{\vec{y_1}, \vec{y_2}, ..., \vec{y_{|Y|}}\}$*, where* $\vec{y_i} = (class_i, count_i)$ *such that* $class_i$ *is the class name and* $count_i$ *is the class count for class* $\vec{y_i}$ *among the* $|\vec{Y}|$ *classes in the representation.*

Algorithm 4 The Class Profile Algorithm

Require: An AP Cluster, C, and its labels
Ensure: The class profile

```
1: for all (Patterns p in an AP Cluster) do
2:    for all (Class Labels l in a Pattern) do
3:       if l.count > maxLabel.count then
4:          maxLabel = l
5:          Assign p.label = maxLabel
6:       end if
7:    end for
8: end for
```

Shannon's Information Entropy for Class Labels Class entropy, H, for a representation is derived from Claude Shannon's work [169]. If a pattern is associated with only one class, then its H is 0, i.e., the best possible score, an indication of certainty. Conversely, if a pattern exists in almost all classes uniformly, then its H is close to 1. Such associations are extended to other representations' H, such as an AP Cluster itself, or an amino acid in a specified column hyperedge of an AP Cluster.

Definition 14 *The H for a representation is computed from the distribution of its class profile and is defined as*

$$H_Y(R) = -\frac{1}{log(|Y|)}\big(\sum_{y_i \in Y} pr(y_i) log(pr(y_i))\big), \tag{5.1}$$

where $|Y|$ is the number of classes and $pr(y_i)$ is the probability of class i occurring in the sequences restricted by that representation, R.

We extend the H for a pattern to other representations (Table 5.3).

Table 5.3: Class Entropy Variables

	Representation R	H Variables
1	AP Cluster	$H_Y(C^l)$
2	pattern	$H_Y(p^i)$
3	column hyperedge	$H_Y(c_j)$
4	distinct amino acid in an column hyperedge	$H_Y(\sigma(c_j)), \sigma(c_j) \in \Sigma(c_j)$

Returning to the text example in Table 5.1, the pattern p^1 =`HELLO*` has a H of $H_Y(p^1) = 0$ with $Y = \{(happy, 5), (sad, 0), (angry, 0)\}$ (Table 5.6). In the same table, the AP Cluster

has a H $H_Y(C^1) = 0.95$ with three classes: $Y = \{(happy, 7), (sad, 3)\}, (angry, 5)\}$.

Table 5.4: Example of an AP Cluster for H of the Horizontal Pattern

	Aligned Columns						Classes			
	c_1	c_2	c_3	c_4	c_5	c_6	happy	sad	angry	$H_Y(p^i)$
p^1	H	E	L	L	O	*	5	0	0	0.00
p^2	M	E	L	L	O	W	2	2	1	0.96
p^3	B	E	L	L	O	W	0	1	4	0.46
$H_Y(C^l)$							7	3	5	0.95

The H for an AP Cluster is obtained horizontally for a pattern, but it is also obtained vertically for column hyperedge. However, in an AP Cluster, the vertical distribution of the class profile is the same for all column hyperedges; therefore, the H for each column hyperedge is the same as that of the AP Cluster. Therefore, we introduce the class information gain (IG) of a column hyperedge in order to measure the change in class information for each column hyperedge when the individual class profiles of the amino acids are taken into consideration.

Amino Acid Class Entropy First, let us begin with the H $H_Y(\sigma(c_j))$ for each distinct amino acid's class profile.

Definition 15 *The amino acid H in the column hyperedge can be expressed as*

$$H_Y(\sigma(c_j)) = -\frac{1}{log(|Y|)}\big(\sum_{y_i \in Y} pr(y_i) log(pr(y_i))\big), \tag{5.2}$$

where $pr(y_i)$ is the probability of class i occurring in the sequences limited by $\sigma(c_j) \in \Sigma(c_j)$ in the column hyperedge c_j.

Algorithm 5 The Amino Acid's Class Profile Algorithm

Require: An AP Cluster and its vertices
Ensure: Each vertex's class profile {Get class profile of amino acid}

```
1: for all (column hyperedges c_j in AP Cluster) do
2:    for all (amino acids σ in c_j) do
3:       for all (patterns p^i in AP Cluster) do
4:          if p^i.column(c_j) = σ then
5:             totalLabel+ = p^i.labelProfile
6:          end if
7:       end for
8:    end for
9: end for
```

Returning to the text example, recall that the set of amino acids in column hyperedge 1 is $\Sigma(c_1) = \{H, B, M\}$; the amino acid class entropies are computed as $H_Y(H(c_1)) = 0.00$, $H_Y(B(c_1)) = 0.46$, and $H_Y(M(c_1)) = 0.96$, indicating respectively in Table 5.5 that H is associated with a unique class; B with two classes; and M with all classes according to their tabulated distribution in each class. Similar to the H of a pattern, when the amino acid belongs to only one class, it has an H of 0; when the amino acid belongs to all the classes uniformly, it has an H of 1.

Class Information Gain of an Aligned Column In information theory, the gain of information is the loss of entropy. Hence, we define the IG of column hyperedge as the loss of entropy when additional information is provided. The entropy of the column hyperedge is obtained based on the class distribution of that column hyperedge, $H_Y(\sigma(c_j))$. If we

Table 5.5: Example showing the H of amino acid

c_j	$\sigma(c_j)$	happy	sad	angry	$H_Y(\sigma(c_j))$
c_1	H	5	0	0	0.00
c_1	B	0	1	4	0.46
c_1	M	2	2	1	0.96
c_2	E	7	3	5	0.95
c_3	L	7	3	5	0.95
c_4	L	7	3	5	0.95
c_5	O	7	3	5	0.95
c_6	*	5	0	0	0.00
c_6	W	2	3	5	0.94

know about the class distributions of each amino acid in the column hyperedge, then we can compute the H and thus, the loss of these entropies is equivalent to IG.

Definition 16 *Let IG be defined as*

$$\Delta H_Y(c_j) = \frac{1}{H_Y(c_j)}\Bigg(H_Y(c_j) - \sum_{\sigma(c_j)\in\Sigma(c_j)} \Big(w(\sigma(c_j))H_Y\big(\sigma(c_j)\big)\Big)\Bigg), \tag{5.3}$$

where $H_Y(c_j)$ is the H of the column hyperedge, c_j (note that $H_Y(c_j) = H_Y(C_l)$), $H_Y(\sigma(c_j))$ is the amino acid H, and $w(\sigma(c_j))$ is the weight for normalizing the occurrences of the amino acid $\sigma(c_j)$ in the column hyperedge c_j. $w(\sigma(c_j))$ is also considered the probability of $\sigma(c_j)$ occurring in $\Sigma(c_j)$ such that $w(\sigma(c_j)) = \frac{count(\sigma(c_j))}{count(c_j)}$.

$$\Delta H_Y(c_1) = \frac{H_Y(c_j) - \sum_{\sigma(c_j)\in\Sigma(c_j)} \Big(w(\sigma(c_j))H_Y\big(\sigma(c_j)\big)\Big)}{H_Y(c_j)} \tag{5.4}$$

$$= \frac{0.95 - \Big(\frac{5}{15}(0.00) + \frac{5}{15}(0.46) + \frac{5}{15}(0.96))\Big)}{0.95} \tag{5.5}$$

$$= \frac{0.48}{0.95} \tag{5.6}$$

$$= 0.50 \tag{5.7}$$

In the text example, the IG of column hyperedge 1 is computed (Table 5.6). The upper bound of IG is 1; $H_Y(\sigma(c_j)) = 1$ is the hypothetical situation of maximum entropy when the amino acids are distributed uniformly among all classes. We observe that the largest IG is 0.5 (in dark grey) as computed in equation 5.4. $H_Y(\sigma(c_j)) = 1$ and the weighted sum of $H_Y(\sigma(c_j))$ is 0 since each amino acid pertains to only one class in that column. The lower bound of IG is 0 when only one amino acid is in that column hyperedge and has the same class distribution as the AP Cluster. Hence, there is no additional gain of information from special associations of the amino acids to the classes.

Class Mutual Information of an Aligned Column Next, we compute the R2 between classes to account for the correlation of the amino acids in the column hyperedge with the class labels.

Table 5.6: Example of an AP Cluster for IG of a Vertical Aligned Column

		Aligned Columns					
		c_1	c_2	c_3	c_4	c_5	c_6
Patterns	p^1	H	E	L	L	O	*
	p^2	M	E	L	L	O	W
	p^3	B	E	L	L	O	W
Amino Acids	$H_Y(\sigma(c_j))$	H	E	L	L	O	*
		0.00	0.95	0.95	0.95	0.95	0.00
		B					W
		0.46					0.94
		M					
		0.96					
IG	$\Delta H_Y(c_j)$	0.50	0.00	0.00	0.00	0.00	0.34

Definition 17 *Let class mutual information be defined as*

$$R2(c_i, Y) = \frac{1}{H(c_i, Y)}\big(H(c_i) + H(Y) - H(c_i, Y)\big), \tag{5.8}$$

where $H(c_i)$ is the amino acid information entropy of the column hyperedge c_i, $H(Y)$ is the class information entropy of the column hyperedge, and lastly, $H(c_i, Y)$ is the joint information entropy between column hyperedge c_i and the class.

Internal Measures without Class Labels

First, the Entropy Redundancy (R1) is a measure that reflects the specificity and diversity of amino acids distributed in a column hyperedge. R1 measures the amino acid variations in a column hyperedge. We adopt two additional cluster validity measures: Normalized

Sum of Mutual Information Redundancy (SR2) [214] and Normalized Sum of Information Gain (SIG) both of which are used to quantify the interdependencies in all the column hyperedges.

To compute the internal measures, we first compute the internal information entropy based on the amino acid distributions in the column hyperedge:

$$H(c_i) = -\frac{1}{log(|\Sigma(c_i)|)}\big(\sum_{\sigma(c_i)\in\Sigma(c_i)} pr(\sigma(c_i))log(pr(\sigma(c_i)))\big), \tag{5.9}$$

with $Pr(\sigma(c_i))$ obtained from counting σ in the column hyperedge, c_j, using the induced data, $\mathbb{D}(C^l)$.

Algorithm 6 Use of induced data to compute cluster validity measures

```
for all (column hyperedges c_i ∈ ℂ = C^l) do
    R1 = column entropies from L_o
    MI = joint entropie from L_o of both
    for all (column hyperedges c_j ∈ ℂ = C^l, where c_j ! = c_i)) do
        SR2 = SUM of all possible joint entropies between (c_i, c_j)
    end for
end for
```

Entropy Redundancy (R1) of a column hyperedge

Definition 18 *The R1 of a column hyperedge c_j, denoted by $R1(c_j)$, is defined as*

$$R1(c_j) = 1 - H(c_i). \tag{5.10}$$

Hence, a conserved column hyperedge has $R1(c_j) = 1$ since the minimum entropy value of

$H(c_j) = 0$. Conversely, if the amino acid occurrences in $\mathbb{D}(C^l)$ are equal-probable, then the maximum entropy value of $H(c_j) = 1$ so that the column hyperedge has $R1(c_j) = 1 - H(c_j) = 0$.

The Normalized Sum of Information Gain SIG of a column hyperedge is the sum of the class information gain obtained from all the pairwise interdependencies of the column hyperedge with the others in the AP Cluster.

Definition 19 *The SIG of an column hyperedge in an AP Cluster is computed as*

$$SIG(c_i) = \frac{1}{n} \sum_{j=1, j \neq i}^{n} IG(c_i, c_j), \tag{5.11}$$

where n is the number of column hyperedges. The IG between all pairs of columns is computed using the induced data of an AP Cluster [113] by

$$IG(c_i, c_j) = \Delta H(c_i|c_j) = \frac{1}{H(c_i|c_j)} \Bigg(H(c_i|c_j) - \sum_{\sigma(c_j) \in \Sigma(c_j)} \Big(w(\sigma(c_j)) H(c_i|\sigma(c_j)) \Big) \Bigg).$$

The Normalized Sum of Mutual Information Redundancy SR2 is formulated as the normalized average of the sum of mutual information redundancy between a column hyperedge and other column hyperedges in an AP Cluster. In other words, SR2 is the sum of all pairwise interdependencies and is computed as mutual information between the current column hyperedges and all the other column hyperedge, for all the column hyperedges in the same AP Cluster.

Definition 20 *The SR2 of a column hyperedge in an AP Cluster is computed as*

$$SR2(c_i) = \frac{1}{n} \sum_{j=1, j \neq i}^{n} R2(c_i, c_j), \tag{5.12}$$

where n is the number of column hyperedges. The mutual information between all pairs of columns is computed using the induced data of an AP Cluster [113] by

$$R2(c_i, c_j) = \frac{1}{H(c_i, c_j)} \big(H(c_i) + H(c_j) - H(c_i, c_j) \big). \tag{5.13}$$

Note that the mutual information of Conserved column hyperedges, that is column hyperedges with only one possible amino acid value, is skipped in the summation. The probabilities are computed from the induced data. The summing of mutual information in SR2 reflects the highest interdependence of a column hyperedge with others, thus reflecting the inherent class dependency characteristics in the induced data of the AP Cluster.

Algorithm 7 Ranking by SR2

Require: An AP Cluster, C, which has the instances of the occurrences from the input
Ensure: Compute the $SR2$ from the data space and rank the top three $SR2$.

```
for all (column hyperedges c_i ∈ ℂ = C^l) do
    R2 = joint entropy from L_o of both
    for all (column hyperedges c_j ∈ ℂ = C^l, where c_j != c_i)) do
        if c_j is not a conserved aligned column, then
            SR2 += mutual information between (c_i, c_j)
        end if
    end for
end for
return average of SR2
```

Returning to the text example(Table 5.1), recall that the set of amino acids in the column hyperedge 1 is $\Sigma(c_1) = \{H, B, M\}$. In Table 5.7, SR2 and R1 are computed from the amino acid in the induced data.

Table 5.7: R1 and SR2 of an AP Cluster for the text example

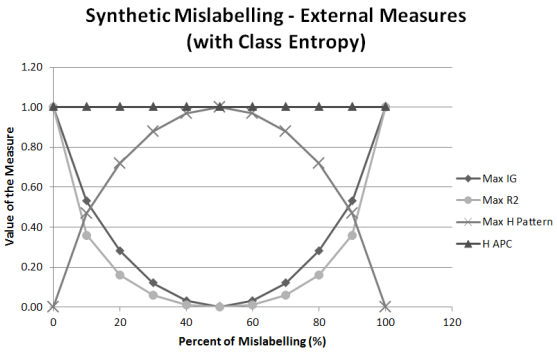
sequence	Aligned Columns					
	c_1	c_2	c_3	c_4	c_5	c_6
s^1	H	E	L	L	O	k
s^2	H	E	L	L	O	u
s^3	H	E	L	L	O	b
s^4	H	E	L	L	O	p
s^5	H	E	L	L	O	w
s^6	M	E	L	L	O	W
s^7	M	E	L	L	O	W
s^8	M	E	L	L	O	W
s^9	M	E	L	L	O	W
s^{10}	M	E	L	L	O	W
s^{11}	B	E	L	L	O	W
s^{12}	B	E	L	L	O	W
s^{13}	B	E	L	L	O	W
s^{14}	B	E	L	L	O	W
s^{15}	B	E	L	L	O	W
$IG(c_j, Y)$	0.5	0	0	0	0	0.34
$R2(c_j, Y)$	0.32	0	0	0	0	0.27
$R1(c_j)$	0	1	1	1	1	0.08
$SR2(c_j)$	0.58	0	0	0	0	0.58
$SIG(c_j)$	0.58	0	0	0	0	1

5.2.2 Synthetic Experiments

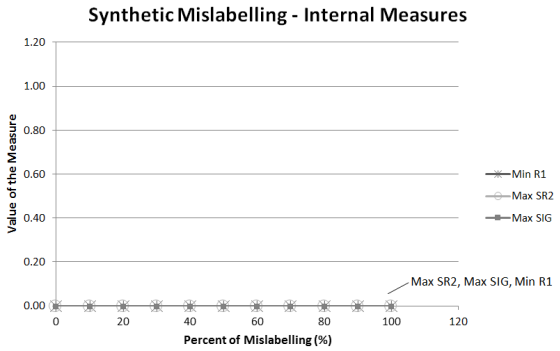
The ability of the proposed external and internal cluster validity measures to avoid bias was shown from experiments conducted on the synthetic datasets that were created to simulate class mislabelling. That is, synthetic datasets that contained class label errors. To evaluate the ability of the external measures to identify the best column hyperedge for class characterization in the protein sequences, we created synthetic datasets that contained variations of amino acids corresponding to different class labels. The degree of mislabelling would reveal in corresponding changes in the external measures. We first created two patterns that differs by one amino acid that is perfectly labelled and change the class label of one sequence occurrence of each pattern for each dataset to simulate mislabelling.

External Measures As the mislabelling increases, both IG and R2 describe the change of the information value will correspond to the degree of mislabelling, showing that external measures are ideal for ranking the quality of the amino acid classifier (Fig. 5.1(a)). As a consequence, the class entropy for patterns increases and then decreases as an indication of the overall increase in randomness when half of the sequences are mislabelled. This same observation is evident for H of the patterns; however, H of AP Cluster remains consistent because the composition of the two patterns did not change.

Internal Measures As for the internal measures (Fig. 5.1(b)), R1, SR2, and SIG they all remain the consistent as the mislabelling increases because they are computed from the induced data alone, and not related to the class labels.



(a) Synthetic Mislabelled External Measures



(b) Synthetic Mislabelled Internal Measures

Figure 5.1: The graphs show the trend of the internal and external measures when there is mislabelling. As mislabelling increases, IG and R2 models it.

5.3 *In Silico* Biological Experiments

To reveal the underlying biological class characteristics of the critical regions, we studied results of *in silico* biological datasets calibrated by their class labels as biologically established ground truth due to homology. Homology describes the shared ancestry; thus, sequence homology describes the sequence ancestry due to speciation event (orthlog) or duplication event (paralog). For example, homologous sequences are orthologous if they descend from the same ancestral sequence by a speciation event, i.e. when a species diverges into two different species. For example, homologous sequences are paralogous due to a duplication event, i.e. when a gene in an organism is duplicated to occupy two different places. Therefore, two types of protein families were studied: the clear gene partitioning of the paralogous SSAT protein family and the ambiguous taxonomic partitioning of the orthologous cytochrome c protein family.

First, for the paralogous SSAT protein family, clear gene function of the proteins are

labelled as SSAT1, SSAT2, and SSAT-L1. Second, for the orthologuous cytochrome c protein family, taxonomic class labels were collected as Mammals, Plants, Insects, and Fungi.

5.3.1 Class Entropy for Patterns and Align Pattern Clusters

To reveal the properties of each cluster validity measures, the AP Cluster corresponding to the distal region of cytochrome c (Table 5.8) was examined in detail. We display the class counts and class entropy of pattern, $H_Y(p^i)$, and AP Cluster, $H_Y(C^l)$. For H, recall that if the representation is associating with only one class, then its H is 0 (shaded in light grey). Conversely, if a the presentation exists in almost all classes uniformly, then its H is close to 1 (shaded in dark grey).

Table 5.8 displays the class distribution and class entropy of the statistically significant AP Cluster that corresponds to the distal binding site, Met, of cytochrome c. The top rows in the table group and align each individual pattern with taxonomic class distribution and class entropy attached to the AP Cluster. The three light grey rows with zero class entropies are patterns 11, 12, and 13, dominated by Mammals. The dark grey rows with the highest class entropies are patterns 3, 5, and 6, which are well distributed throughout all the taxonomic classes. The distal AP Hypergraph has class entropy 0.95 (in dark grey), indicating the class distribution for the AP Cluster is almost evenly across the classes.

Discussion Here, we observe that the AP Clusters may contain patterns with low entropy, clearly associated with specific classes, and also high entropy patterns, shared by

Table 5.8: Class Entropies of Patterns in the Distal APC found in Cytochrome C

	AP Cluster	Mammals	Plants	Fungi	Insects	$H_Y(p^i)$
p^1	TLYDYLL*NP******************	0	21	1	0	0.13
p^2	*LFEY*LENPKK****************	0	1	5	9	0.6
p^3	********NPKKYIPGTKM*********	30	24	20	10	0.95
p^4	********NPKKYIPGTKMVF*******	0	23	1	4	0.4
p^5	****Y*LENPKKYIPGTKM*********	30	1	16	10	0.77
p^6	***EY*LENPKKYIPGTKM*********	30	1	5	9	0.66
p^7	********NPKKYIPGTKMAFGGLKK**	0	1	17	0	0.15
p^8	*LYDYLL*NPKKYIPGTKMVF*******	0	21	1	0	0.13
p^9	***EY*LENPKKYIPGTKMIFAG*****	22	0	0	6	0.37
p^{10}	****Y*LENPKKYIPGTKMAFGGLKK**	0	1	15	0	0.17
p^{11}	TLMEY*LENPKKYIPGTKMIF*******	30	0	0	0	0
p^{12}	TLMEY*LENPKKYIPGTKMIFAGIKK*K	17	0	0	0	0
p^{13}	TLMEY*LENPKKYIPGTKMIFAGIKK**	20	0	0	0	0
p^{14}	TLYDYLL*NPKKYIPGTKMVFPGLKKPQ	0	18	1	0	0.15
p^{15}	****YLL*NPKKYIPGTKMVFPGLKKP*	0	22	1	0	0.13
$H_Y(C^l)$	Distal AP Cluster Total	30	25	21	10	0.95

more classes. Each individual pattern with H of 0.00 may be a good classifier, but once it is grouped and aligned into an AP Cluster, the collective AP Cluster share the class distributions with resulting H close to 1 (in dark grey). Hence such AP Cluster as a whole is poor classifier; thus, we need to identify the amino acids at the column hyperedges to partition the classes more precisely.

5.3.2 Class Entropies for Amino Acids

To explore how the amino acids in each of the vertical column hyperedges associate with classes, we display their class counts and class entropy, $H_Y(\sigma(c_j))$. The column hyperedge, c_{21}, has four possible amino acids:

- The wild card * with the H of 0.90, indicating that it pertains in more than one class, actually, in all four classes Mammals, Plants, Fungi, and Insects;
- amino acid A with a medium H of 0.37, indicating that it pertains to more than one class, i.e., two classes Mammals and Insects;
- amino acid P with a low H of 0.13, indicating that it pertains to mostly one class, i.e., Plants; and
- amino acid G with a low H of 0.15, indicating that it also pertains mostly to one class, i.e., Fungi.

The four distinct amino acids together in column hyperedge 21 produced an IG of 0.61, indicating a considerable gain of class information of that aligned column when class entropy and its amino acids are considered.

Table 5.9: Amino Acids in aligned column 21 of the Distal AP Cluster

$\sigma(c_j)$	Mammals	Plants	Fungi	Insects	$H_Y(\sigma(c_j))$
*	8	2	3	4	0.90
G	0	1	17	0	0.15
A	22	0	0	6	0.37
P	0	22	1	0	0.13
$IG(c_{21})$					0.61

Discussion In general, it is easy to identify the amino acid with the best or the worst Class Entropy. However, the individual amino acids alone cannot describe the class characteristics inherent in the full AP Cluster, therefore we need to look at each aligned column. The IG of a column hyperedge renders a more effective cluster validity measure that computes the change in the Class Entropy from that of the AP Cluster after considering each amino acid in that column hyperedge. Therefore, IG of an column hyperedge is used because it is a better overall class measure.

5.3.3 Class Information Gain and Class Mutual Information for the Column Hyperedges

From this point forward, the cluster validity measures are applied to the representation of column hyperedge. First, we examine the Class Information Gain and Class Mutual Information for column hyperedge representations. In the distal AP Cluster of cytochrome c (Table 5.10), the maximum IGs and R2s are column hyperedge 73, 90, and 92 (in dark grey), each with multiple amino acids in the column hyperedges. High IG and R2 values

indicate multiple amino acids in the column hyperedge have the most gain in class entropy is gained after the amino acid class entropies are considered together. Conversely, the minimum IGs and R2s are column hyperedges 79 to 80 (in light grey) of 0.00, indicating no additional class entropy is gained vertically when considering the lone amino acid from the full AP Cluster. Additionally, column hyperedge 77 has R2 of 0.00, but not IG, due to calculation from pattern space for IG but from induced data for R2 and the rest of the column hyperedge measures.

Discussion The column hyperedges with multiple amino acids that have even class distribution have high IG and R2. Therefore, the amino acid variations with the highest IG and R2 provide the best classifier for partitioning the classes. IG and R2 shows the same trends, which is also demonstrated later in the synthetic dataset and the large-scale comparisons.

5.3.4 Internal Measures

To demonstrate the trends and relationships between the internal measures, which do not depend on external class labels, we examined the relationship between internal measures R1, SR2, and SIG. From the set of induced data, the internal measures do not depend on class labels and are computed on data along.

High SR2 corresponds to a high SIG (in dark grey), indicating strong interdependence of that column hyperedge with all other column hyperedges. In order to minimize the effect of conserved aligned columns, which has R1 of 1.00 (in dark grey), its SR2 and SIG

Table 5.10: All Measures for selected aligned columns in the Distal AP Cluster for Cytochrome C

	c_{71}	c_{72}	c_{73}	c_{74}	c_{75}	c_{76}	c_{77}	c_{78}	c_{79}	c_{80}	c_{81}	c_{82}	c_{83}	c_{84}	c_{85}	c_{86}	c_{87}	c_{88}	c_{89}	c_{90}	c_{91}	c_{92}	c_{93}	c_{94}	c_{95}	c_{96}	c_{97}	c_{98}
p^1	T	L	Y	D	Y	L	L	*	N	P	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
p^2	*	L	F	E	Y	*	L	E	N	P	K	K	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
p^3	*	*	*	*	*	*	*	*	N	P	K	K	Y	I	P	G	T	K	M	*	*	*	*	*	*	*	*	*
p^4	*	*	*	*	*	*	*	*	N	P	K	K	Y	I	P	G	T	K	M	V	F	*	*	*	*	*	*	*
p^5	*	*	*	*	Y	*	L	E	N	P	K	K	Y	I	P	G	T	K	M	*	*	*	*	*	*	*	*	*
p^6	*	*	*	E	Y	*	L	E	N	P	K	K	Y	I	P	G	T	K	M	*	*	*	*	*	*	*	*	*
p^7	*	*	*	*	*	*	*	*	N	P	K	K	Y	I	P	G	T	K	M	A	F	G	G	L	K	K	*	*
p^8	*	L	Y	D	Y	L	L	*	N	P	K	K	Y	I	P	G	T	K	M	V	F	*	*	*	*	*	*	*
p^9	*	*	*	E	Y	*	L	E	N	P	K	K	Y	I	P	G	T	K	M	I	F	A	G	*	*	*	*	*
p^{10}	*	*	*	*	Y	*	L	E	N	P	K	K	Y	I	P	G	T	K	M	A	F	G	G	L	K	K	*	*
p^{11}	T	L	M	E	Y	*	L	E	N	P	K	K	Y	I	P	G	T	K	M	I	F	*	*	*	*	*	*	*
p^{12}	T	L	M	E	Y	*	L	E	N	P	K	K	Y	I	P	G	T	K	M	I	F	A	G	I	K	K	*	K
p^{13}	T	L	M	E	Y	*	L	E	N	P	K	K	Y	I	P	G	T	K	M	I	F	A	G	I	K	K	*	*
p^{14}	T	L	Y	D	Y	L	L	*	N	P	K	K	Y	I	P	G	T	K	M	V	F	P	G	L	K	K	P	Q
p^{15}	*	*	*	*	Y	L	L	*	N	P	K	K	Y	I	P	G	T	K	M	V	F	P	G	L	K	K	P	*
T	L	Y	D	Y	L	L	*	N	P	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	
	0.55	0.87	0.13	0.13	0.95	0.12	0.95	0.3	0.95	0.95	0	0	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.41	0.41	0.9	0.9	0.86	0.86	0.86	0.83	0.95
	*	*	F	E	*	*	*	E			K	K	Y	I	P	G	T	K	M	V	F	G	G	L	K	K	P	K
	0.67	0.4	0.62	0.68	0	0.78	0	0.77			0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.4	0.95	0.15	0.93	0.49	0.79	0.79	0.13	0
			*	*																A		A		I				Q
			0.4	0.42																0.15		0.37		0				0.15
			M																	I		P						
			0																	0.33		0.13						
H	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95
IG	0.38	0.19	0.76	0.49	0.04	0.38	0.04	0.35	0.00	0.00	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.67	0.03	0.61	0.03	0.49	0.15	0.15	0.32	0.39
R2	0.06	0.15	0.58	0.29	0.04	0.36	0.00	0.36	0.00	0.00	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.58	0.05	0.42	0.03	0.38	0.12	0.12	0.36	0.29
R1	0.76	0.66	0.17	0.46	0.84	0.32	1.00	0.32	1.00	1.00	0.91	0.91	0.84	0.84	0.84	0.84	0.84	0.84	0.84	0.08	0.55	0.01	0.24	0.04	0.13	0.13	0.08	0.12
SR2	0.06	0.09	0.16	0.13	0.06	0.17	0.00	0.17	0.00	0.00	0.16	0.16	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.2	0.11	0.21	0.12	0.19	0.15	0.15	0.18	0.15
SIG	0.39	0.4	0.24	0.29	0.35	0.41	0.00	0.36	0.00	0.00	0.43	0.43	0.45	0.45	0.45	0.45	0.45	0.45	0.45	0.27	0.3	0.27	0.27	0.28	0.29	0.29	0.27	0.24

are automatically set to 0.00

Discussion When examined in detail, class partitioning of an AP Cluster and its induced data are both reflected by the amino acid variations and conservations correlated with shared class functionality across the class labels. Both SR2 and SIG in this AP Cluster demonstrate that Mammals and Insects may have shared functionalities since both are from the Animalia (also called Metazoa) kingdom. It is important to note that functionality and class may not be partitioned in the same manner. In all our experiments, we found that SR2 and SIG are good classifiers in the sense that they associate amino acids with not only distinct classes but also across multiple classes, indicating shared functionalities.

5.3.5 Top Ranking Aligned Pattern Clusters

To confirm that the cluster validity measures are appropriate for identifying column hyperedges as classifiers, we rank the AP Clusters by their statistical significance and examine each of their optimal cluster validity measures (Table 5.11).

Discussion Each of the top ten statistically significant AP Clusters are listed with the optimal value of each measures and its column hyperedges containing that value. The multiple column hyperedges per optimal value is due to more than one column hyperedge associated with that value; thus, all those column hyperedges are listed on multiple rows. We observe that the optimal internal measures tend to correspond to the same column hyperedges and the optimal external measures also tend to correspond to the same column hyperedges, thereby implying a linear correlation between the internal measures.

Table 5.11: The Top 10 AP Hypergraphs of the Cytochrome C Ranked by Statistical Significance

Rank	AP Cluster	Min R1	Posn R1	Max SR2	Posn SR2	Max SIG	Posn SIG	Max IG	Posn IG	Max R2	Posn R2
1	MGDVEKGKKI FV[QK]T[RK] CAQCHTV[ED] KGGKHKTGPNL HGLFGRKTGQA PG	0.01	24K 25G 26G	0.42	35H 36G 37L 38F 39G 40R 41K 42T 43G 44Q 45A	0.81	1M 4V	0.46	13[QK]	0.52	13[QK]
2	KGAGKHK[QT] GPNL[HN]GL FGR[KQ][TS] AG[TQ][QT] [AT]PG[YF]SYS	0.05	22[TQ]	0.23	22[TQ]	0.64	3A	0.56	23[QT]	0.55	8[QT]
3	TL[YFM][DE] YLLENPKKYI PGTKM[VAI] F[GAP]G[LI] KKP[KQ]	0.01	22[GAP]	0.3 17T	13Y 14I 15P 16G 18K 19M	0.45 17T	13Y 14I 15P 16G 18K 19M	0.76	3[YFM]	0.58	3[YFM] 20[VAI]
4	AG[YF]SY[TS] [DA]ANKNKG ITWGE	0.02	12K	0.27	15T 16W	0.57	1A	0.52	13G	0.52	13G
5	ERADLIAYLK KATNE	0.02	11K	0.34	7A 8Y 9L 10K	0.49	7A 8Y 9L 10K	0.42	12A 13T	0.39	15E
6	MGDVEKGKKI FVQ	0.12	12V 13Q	0.18	10I 11F 12V 13Q	0.4	3D	0.42	1M	0.58	1M
7	G[KA]GHK[TQV] GPNL[NH]G	0.01	11[NH]	0.53	2[KA]	0.81	3G	0.73	6[TQV]	0.74	3G
8	GEK[LI]FKT [RK]CA	0.00	4[LI] 10A	0.64	4[LI] 8[RK] 10A	0.88	3K	0.72	3K 4[LI] 8[RK] 10A	0.56	2E 4[LI] 8[RK] 10A
9	AGYSY[ST]DA	0.09	7D 8A	0.48	7D 8A	0.74	7D 8A	0.51	6[ST]	0.41	7D 8A
10	GYLKK[AP] [TQ]	0.00	2Y 6[AP]	0.82	2Y 6[AP]	1.00	1G 2Y 6[AP]	0.57	7[TQ]	0.51	7[TQ]

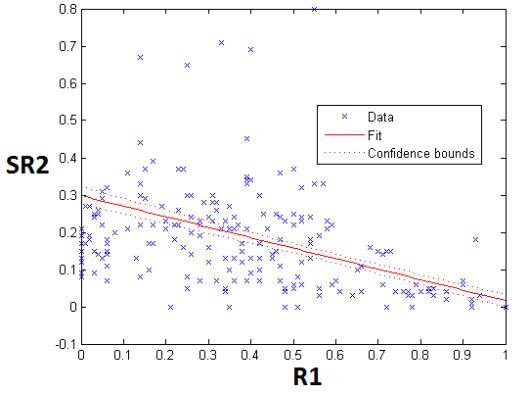
We recommend using top ranking internal measures for finding characterizing amino acid variations for partitioning the classes.

5.3.6 Relationship between Cluster Validity Measures

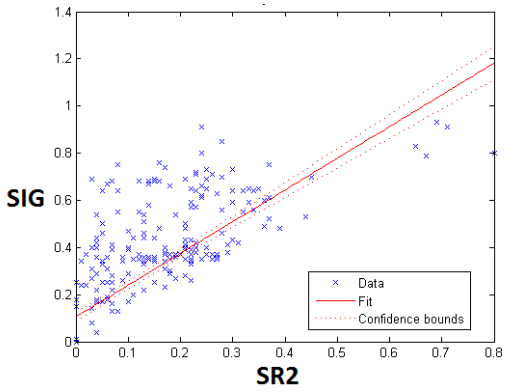
To study the relationships amongst class characteristics with internal measures, we compared experimental results obtained from two *in silico* biological datasets. The internal measures are compared to one another, and the external measures are compared on two different types of collected external class labels.

Internal Measures The internal measures, R1, SR2, and SIG, do not require external class labels, and the external measures, do. To determine the relationship between the internal measures, between the three internal measures, we studied the linear regression by plotting the linear correlation and calculating the degree of error, which is the coefficient of determination (R^2). Note that this is different from R2, which unfortunately has the same symbol. The two different types of class labels were not considered for internal measures because these measures depend on data alone and thus are not affected by differences in the class labels. Using linear regression, R1 shows a negative relationship with SR2 and SIG, whereas SR2 and SIG are positively correlated (Fig. 5.2).

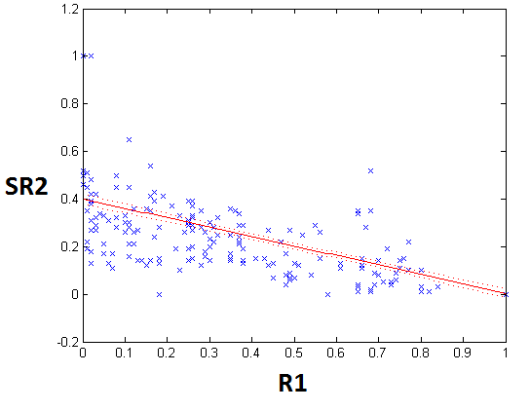
This correlation is because an AP Cluster is not an arbitrary array of amino acids, but rather, a horizontally aligned array of statistically significant patterns (i.e., amino acid associations). Hence, any differences in amino acids, the amino acid associations extracted from patterns, in the column could have a strong interdependence with amino acids in other



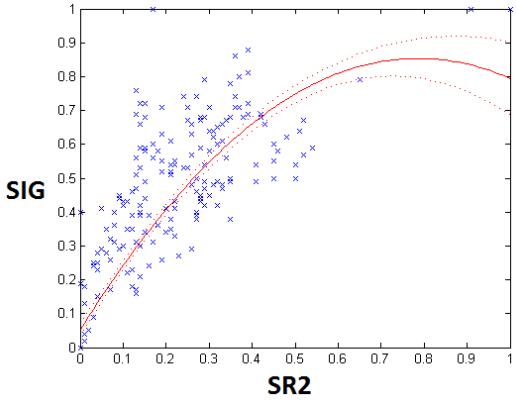
(a) CytoC: SR2 v.s. R1



(b) CytoC: SR2 v.s. SIG



(c) SSAT: SR2 v.s. R1



(d) SSAT: SR2 v.s. SIG

Figure 5.2: The negative linear relationship between R1s and SR2s for each column hyperedge in the AP Cluster. The linear relationship between SIGs and SR2s indicates that the two different formulae evaluate similar behavior in the class labels.

aligned columns, explaining why low R1 corresponds to high SR2. An aligned column with low R1 suggests that the amino acids are more diverse.

SR2 and SIG have the strongest correlation (Fig. 5.3), especially for SSAT, implying that this dataset have clean class partitioning, which leads to strong correlation between them. Furthermore, such correlation is stronger in the clean paraloguous protein family than in the ambiguous orthologuous protein family, indicating that correct partitioning yields stronger correlations between the internal measures.

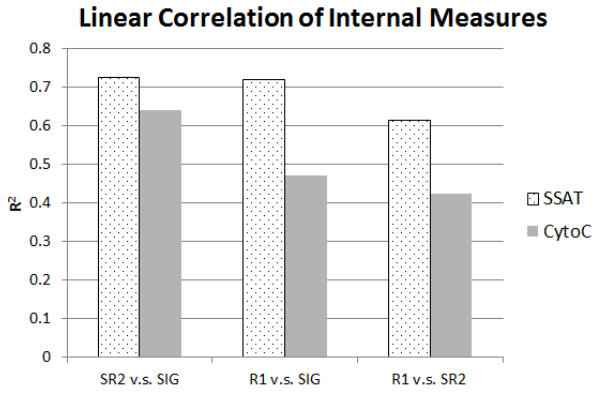
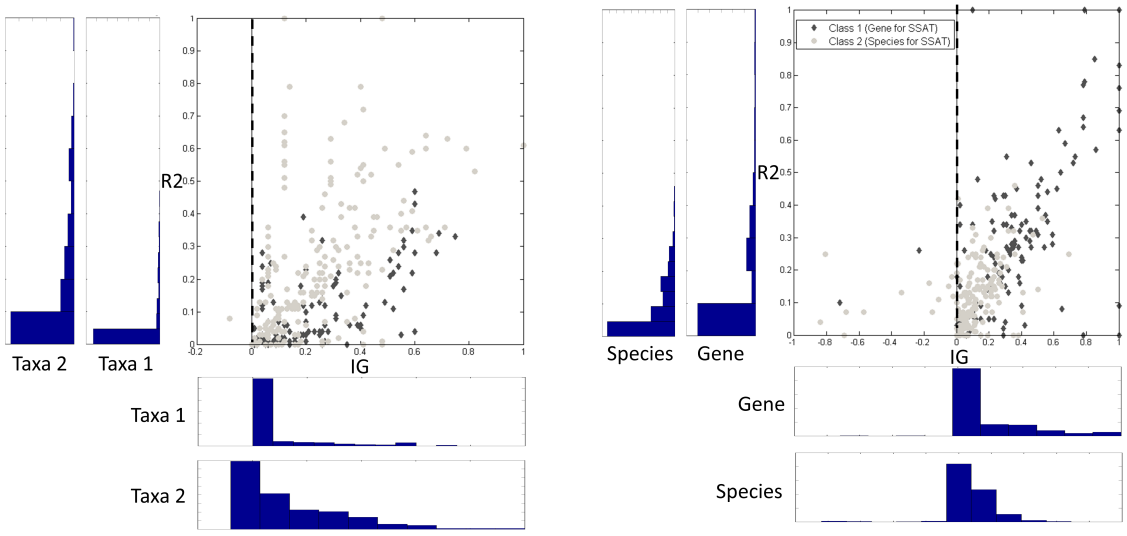


Figure 5.3: The strength of R^2 measures the relationship between the internal measures. SR2 and SIG have the strongest correlation, and paraloguous proteins have a stronger correlation than orthologuous proteins.

External Measures To determine the effects of different class label characteristics on external measures, we compared R2 and IG of each datasets with its unique class labels and plotted their histograms (Fig. 5.11). Because the external class labels are incorporated into the formulae for external measures, two different types of external class labels for the same set of protein sequence data were collected. First, for the paralogous SSAT

protein family, two sets of class labels were collected. The first set of class labels correctly describes the function of the proteins as SSAT1, SSAT2, and SSAT-L1. The second set of class labels partition the sequences incorrectly by their taxonomic species. Second, for the orthologuous cytochrome c protein family two different types of taxonomic class labels were collected. The first set of class labels using kingdoms consists of the top level in Uniprot taxonomy and the second set consists of second level in Uniprot taxonomy which partition the sequences into more number of classes.



(a) CytoC: IG v.s. R2

(b) SSAT: IG v.s. R2

Figure 5.4: For each protein, IG v.s. R2 were plotted for each of the two different types of class labels. Their corresponding histogram values are bar graphs along the x-axis and y-axis.

The results showed a weak linear relationship between IG and R2. We observe that, when the taxonomy is expanded to increase the number of classes, the histograms of both measures expand outward, indicating an increase of the measures, thus the scatter plot

expands as well. This observation indicates that, as the number of taxonomic classes increases, it is less likely that a representation belongs to only one class, and thus, class entropy is smaller than it was previously and is less likely to be zero. Therefore, as shown in the histograms, R2 is also always smaller and the value of IG increases, leading to a more expanded and larger value of IG. IG differences the two different histograms for SSAT indicate that more information can be gained when there are more classes, such that the results can more likely be divided into negative IG. Therefore, negative IG implies incorrect partitioning as observed in the SSAT taxonomy IGs.

We recommend using the external measures to rank the quality of the class labels being used, as well as the internal measure for ranking aligned columns internally by data alone.

5.3.7 State-of-the-Art Comparisons

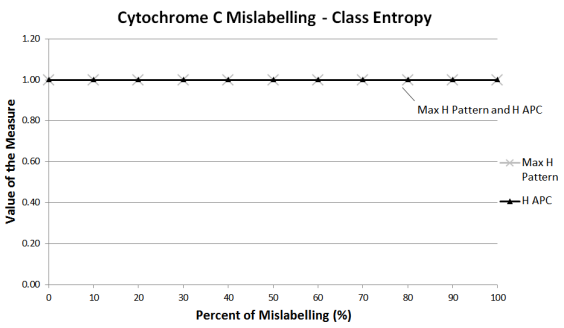
Cytochrome c and SSAT datasets were used to compare the precision of our method in revealing mislabelling to those of the other classification algorithms: SVM and HMM (Fig. 5.5 and 5.6). The cytochrome c protein demonstrates noisy partitioning using taxonomic class labels, while the SSAT protein demonstrates clean partitioning using gene class labels. The Uniprot cytochrome c dataset was limited to only Mammals and Plants; the ENSEMBL SSAT dataset was limited to only SSAT1 and SSAT2. Each dataset was separated into a training set and a testing set. For each set of experiments, the class labels for the training set were altered to imitate mislabelling and the entire testing set was kept constant. Both SVM and HMM are classification learning algorithms that use the training set to accurately predict the testing set. However, our method is a clustering algorithm

with external and internal cluster validity measures designed to reveal class characteristics. Thus, both the training set and the testing set were used to identify the best column hyperedge for each measure as summarized in the conclusion.

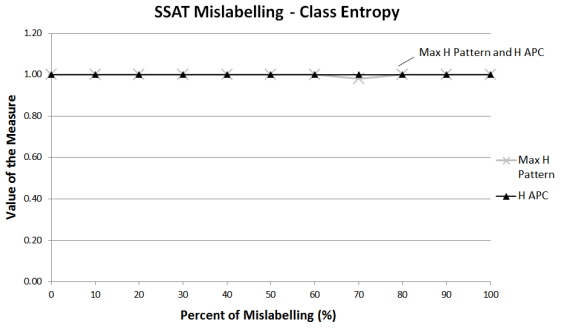
For SVM, the Shogun package was executed using linear hard-margin SVM with local alignment kernel. For HMM, the input sequences were first aligned using Matlab Bioinformatics Toolbox and then the profile HMM was built using HMMER's hmm build.

Mislabelling Class Labels Classification is strongly influenced by errors in the training set since the learned model will propagate the mistakes. Therefore, to show the effect of the errors in the training set, class labels in the training set are changed to imitate mistakes, and the accuracy of the predicted class labels of the testing set is measured (Fig. 5.5(a) and 5.5(b)). The accuracy of the classification algorithms, SVM and HMM, drops significantly as mislabelling increases. For our method, SR2 does not change since it is independent of the class labels. IG drops and climbs to demonstrate that it can independently assess the associations between the data and the class labels. It should be noted that only the mislabelled training set was used for IG to demonstrate the concave behaviour of IG in measuring the proportion of mislabelling. When mislabelling reaches 100%, i.e., becomes all errors, the IG increase again as it reverses in the error rate.

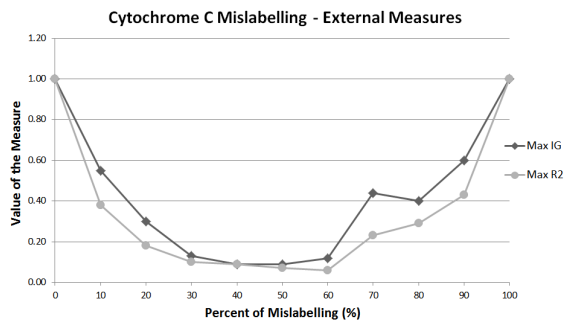
Unbalanced Distribution of Training Sets To study the effect of the size of the training sets and unbalanced class labels on the accuracy of the trained models, the input training set is manipulated and the accuracy of the trained model is measured. In the training sets, the number of Mammals remains constant, whereas the number of Plants



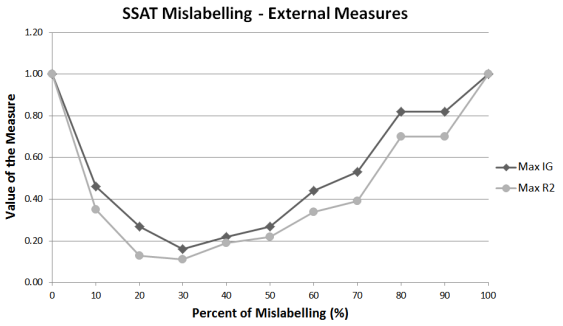
(a) Cytochrome C Mislabelling – External Measures



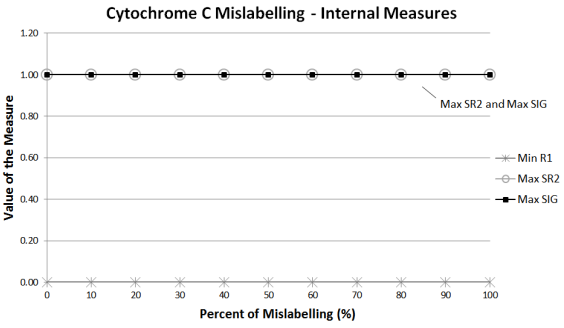
(b) SSAT Mislabelling – External Measures



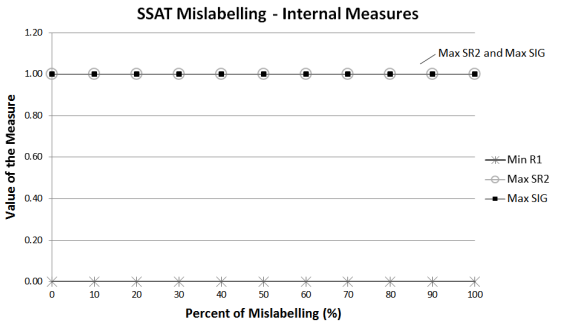
(c) Cytochrome C Mislabelling – External Measures for column hyperedge



(d) SSAT Mislabelling – External Measures for column hyperedge

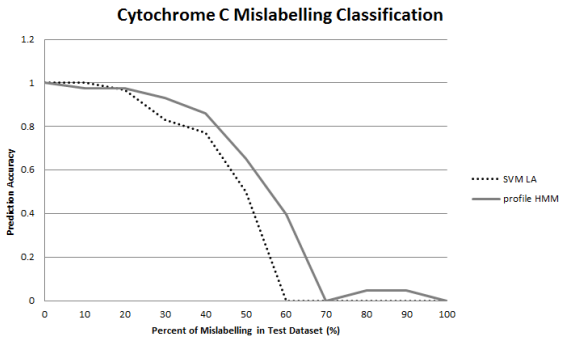


(e) Cytochrome C Mislabelling – Internal Measures

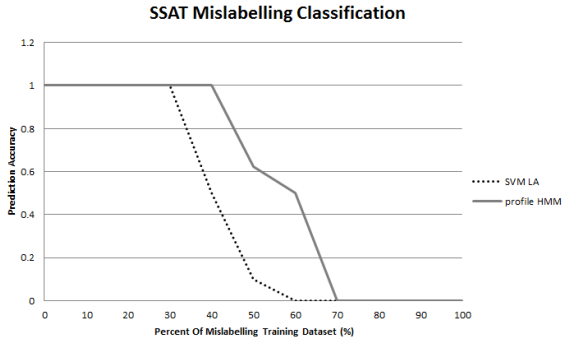


(f) SSAT Mislabelling – Internal Measures

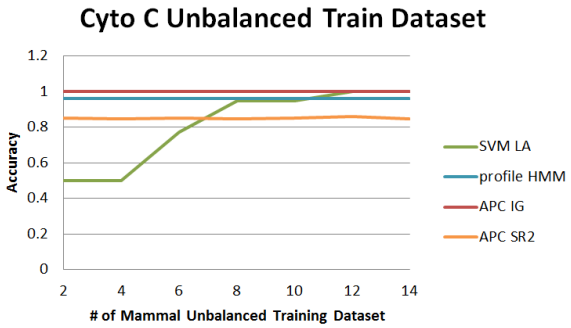
Figure 5.5: The graphs show our internal and external measures for cytochrome c and SSAT for class mislabelled training sets.



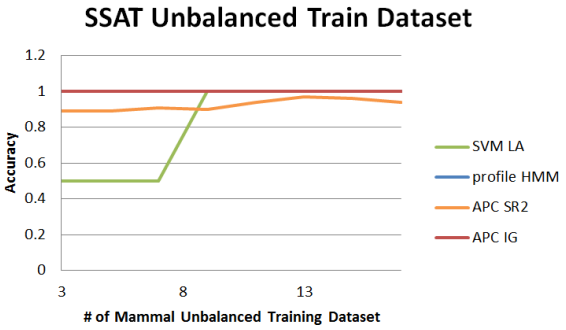
(a) Cytochrome C Mislabelling – Supervised Methods



(b) SSAT Mislabelling – Supervised Methods



(c) Cytochrome C Unbalanced Training Set



(d) SSAT Unbalanced Training Set

Figure 5.6: The graphs show the comparison of our method with HMM and SVM for cytochrome c and SSAT for class mislabelling and unbalanced training sets.

is varied (Fig. 5.6(c) and 5.6(d)). SVM is unable to accurately predict the testing set when there is insufficient training data for the Plant class. However, our cluster validity measures and the profile HMM are not influenced by the size of the training sets because IG is normalized based on the size of the classes and the profile HMM is trained based on the MSA of the entire input sequences.

Runtime Comparisons The runtimes for our AP Cluster, SVM and HMM for the two biological datasets are compared (Table 5.12). Results show that SVM is fast but inaccurate when insufficient training data is available. On the other hand, HMM is accurate, as it is unaffected by size of the training set, but is time consuming because it first build the MSA for training. Therefore, our AP Clusters are accurate as well as fast with a faster runtime than SVM and MSA.

Table 5.12: Runtime Comparisons (in seconds)

In Seconds	Cyto C	SSAT
AP Cluster (No training)	3.46	0.92
SVM Training	4	1
SVM Testing	1	1
HMM Training	20	20
HMM Testing	30	10

5.4 Chapter Conclusions

AP Clusters allow the effective use of clustering that is more general and unbiased than the traditional classification algorithms, such as SVM and HMM, since AP Clusters depend

only on internal data alone without requiring external class labels in the algorithm design. For the synthetic dataset and *in silico* comparison to SVM and HMM, we observed that internal measures are unaffected by mislabelling, whereas external measures describe the relationships. A large-scale experiment of the cluster validity measures of column hyperedges was conducted to compare (1) the relationship between internal measures and (2) the behavior of external measures to different class labelling. Finally, the superiority of cluster validity measures over SVM and HMM classifiers in handling labelling errors was confirmed.

Cluster validity measures reveal class characteristics that are inherent in the highly functionally correlated AP Clusters. Hence, AP Clusters relate regional functionalities to inherent class partitioning more effectively for both external and internal measures. If known, the class labels can be used as external validations, whereas if absent, other internal measures can be used for revealing class discriminability. Therefore, AP Clusters are suitable for dealing with complex and large datasets because the simple assumption of class labelling cannot be justified when applied to the entire set of data in an unrestricted manner.

In conclusion, amino acid variations manifested in column hyperedges with optimal cluster validity measures are more likely to be functionally significant. Biochemists who study protein functionalities should focus on these significant amino acids rather than enumerate all possibilities in high-throughput experiments. Currently, in the next chapter, we are studying the co-occurrence of AP Clusters and their column hyperedge variations.

Chapter 6

Co-Occurrence Clusters of Aligned Pattern Clusters

6.1 Chapter Introduction

Identifying functional regions on proteins is essential for understanding biological mechanisms and for designing new drugs. Due to the accessibility to protein sequences on the web, it is more effective to look for conserved segments from a set of functionally similar protein sequences than to perform laborious and time-consuming experiments and computationally intensive modeling. The study of conserved functional regions relies on the assumption that amino acids in functional regions are integral and thus undergo fewer mutations throughout evolution than less functionally important amino acids [124]. Therefore, the functional regions of protein structures can be obtained from analyzing protein

sequences that have similar biological functions.

Multiple sequence alignment (MSA) [186, 139] is a traditional computational method which is capable of aligning homologous protein sequences that are highly similar. However, it is unable to discover functional regions in more divergent protein sequences. Consequently, MSA is a global alignment method suitable for studying closely related proteins but not proteins that have only region-wise, partially functional similarities [187]. It has also been shown that finding the global optimal alignment is an NP-complete problem [203]. Coupling analysis [204, 35, 135] is a method based on MSA that examines the substitution correlation between two aligned columns within the MSA. This study hypothesizes that if two residues form a contact within a protein, then an amino acid substitution at one position is expected to be compensated for by a substitution in another position over the evolutionary time-scale. This observation suggests that co-occurring residues on the same protein can provide insight into the protein's structure. However, due to the dependence on MSA and the complexity of the method, determining the underlying statistical model requires a large number of homologous non-redundant protein sequences. Evolutionary tracing [124] is another method based on clustering alignments. The consensus within and across each group is identified to allow the study of divergent residues that are globally or functionally preserved in a protein family. Once again, evolutionary tracing is based on full sequence similarity requiring mutagenesis information for clustering [129]. Hence, it is not effective for revealing local functionality. Both coupling analysis and evolutionary tracing are based on examining pairwise amino acid correlations from MSA which focuses on two identified sites and does not take into account other sequence information.

In comparison to traditional methods, our algorithm finds and analyzes higher order

sequence patterns in conserved regions, improving the capacity to reveal cross pattern association and local and distant functionality. In our previous work, we introduced Aligned Pattern Clusters (AP Clusters) [115] to represent functional regions as an alternative to position weight matrices [221]. Aligned Pattern Clusters are sequence patterns with variations and conservation without assuming independence between residues [115] at sites. Its strength lies in the retention of statistical significance along the amino acids on a sequence and also the tracking of distribution of their occurrences across the sequences. With this novel representation, we are now able to exploit the APC occurrences and study the co-occurrence between their patterns on the same protein sequence.

We hypothesize that co-occurring patterns reflect the joint functionality that are needed for co-operative biological functions such as chemical bonds or binding sites. Thus, we address the following two research questions: 1) Given a set of homologous protein sequences, how can frequently co-occurring patterns be efficiently discovered? 2) How can the biological reasoning and significance of these co-occurrences be confirmed? To test these hypotheses, we used our co-occurrence clustering algorithm to find highly co-occurring patterns among a cluster of APCs and then studied their biological functions. First, we collect homologous protein sequences from the protein databases Pfam [63] and UniProt [17] as input. Next, we design an efficient algorithm based on our previous work [218, 115] to find and represent the frequently co-occurring patterns. Finally, we verify our results by comparing the three-dimensional distance between the co-occurring patterns against the average distance between the regions spanned by the patterns. To confirm the biological functions of the co-occurrences, we search the related scientific literature to support the conceived role of these co-occurring patterns.

In view of the above mentioned computational results and biological observations accomplished in this paper, the contributions of this study mirror the answers to the research questions in two ways. First, we have established an algorithm that discovers co-occurring functional regions that are statistically reliable, measurable, and efficient. To our knowledge, this study is the first to identify the co-occurrence of patterns rather than residues. Compared to existing algorithms used to study correlations in amino acid residues, the novelty of our algorithm is that it does not require a large number of homologous protein sequences to identify pattern co-occurrences. Secondly, we have verified these co-occurrences by using the co-occurring patterns' three-dimensional closeness and by searching biological literature for support, enriching our understanding of the underlying mechanism. Novel co-occurrence relationships will provide new insight for the biological community for use in their study on protein functionalities.

Previously, the Chapter 2 of this dissertation shows that Aligned Pattern Clusters (AP Clusters) are able to provide a knowledge-rich representation of functional regions of a protein. With this novel representation, this chapter tries to show that joint functionality of distant regions in a protein can also be revealed by the co-occurrence of patterns contained in distant AP Clusters discovered on the same proteins. In other words, in this chapter we attempt to study and exploit the notion of co-occurrence of patterns in distant AP Clusters discovered on the same protein in order to find out how co-occurring AP Clusters are able to reveal interacting or binding segments within a protein.

The three contributions in this chapter of the dissertation are:

- A framework to study functional regions of proteins by exploiting the co-occurrences

of patterns to reveal concurrent distant functions and structural relations.

- An algorithm which is statistically reliable, efficient, and visualizable (in domain location, structural and functional relation, amino acid conservation and variations) in an integrated process and manner.

- Those discovered co-occurrence of patterns that are novel to the biological community will provide new insights to their studies of biological functions.

The dissertation chapter is organized as follows: methodology section describes details of the proposed clustering and the three-dimensional confirmation; the experimental section provides the results and discussion to *In Silico* case study and structural and functional biological significance; and the chapter conclusion is the concluding remarks.

6.2 Methods

6.2.1 Algorithm definition and details

The methodology proposed in this dissertation chapter combines three algorithms together to obtain the Co-occurrence Cluster of Aligned Pattern Clusters (Co-occurrence Cluster) (Fig. 6.1). The first two algorithms are adopted from our previously dissertation chapters: 1) a pattern discovery algorithm that discovers statistically significant sequence patterns from a set of sequences of a protein family while pruning the redundant patterns [218]; 2) an Aligned Pattern Cluster (AP Cluster) algorithm that obtains compact aligned groups of statistically significant patterns referred to as AP Clusters. These AP Clusters contain

variations with adjustable low information entropy [115]. Finally, in the third and main contribution algorithm of this dissertation chapter , Co-occurrence Clusters are obtained by clustering the AP Clusters discovered using spectral clustering [197] with a co-occurrence score adopted as a measure of distance.

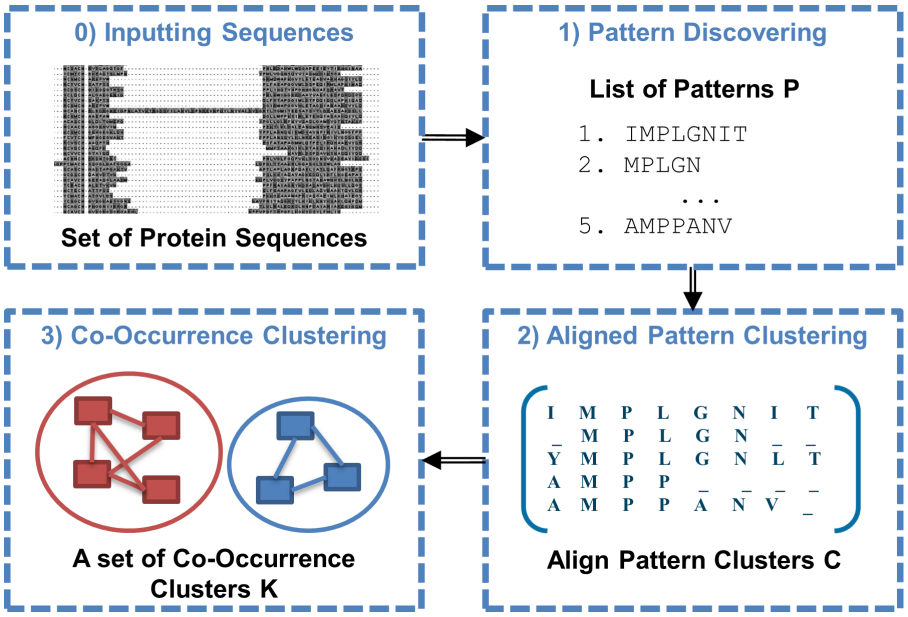


Figure 6.1: The overall process of our methodology is represented by a pipeline consisting of three algorithms.0) the input is a set of sequences from the same protein family; 1) the published pattern discovery algorithm, which results in a list of patterns; 2) the published APC algorithm, which results in a set of APCs; and 3) the new Co-Occurrence Cluster algorithm, which cluster APCs by their co-occurrence scores.

Clustering AP Clusters to Co-occurrence Clusters

Co-existence of patterns in different locations of the same protein may indicate that they are functionally related and important for the protein family. In Co-occurrence Clusters,

we first apply a spectral clustering algorithm to cluster AP Clusters using a co-occurrence score between AP Clusters as the similarity measure. Let the graph $G = (V, E)$ be a relationship graph with AP Clusters as vertices. Let each vertex v be an AP Cluster, and let each weighted edge e be the co-occurrence for two AP Clusters; the edge weight is the co-occurrence score to be defined later between the two APCs. The spectral clustering algorithm is used to obtain Co-occurrence Clusters based on the co-occurrences between the AP Clusters.

Co-occurrence score To tell how many patterns out of the total number of the discovered patterns co-occur in two AP Clusters, we need a co-occurrence score which will be used as the similarity measure for clustering co-occurrent AP Clusters. The co-occurrence scores quantify how often patterns in two AP Clusters appear together on the same sequence. The Jaccard index is adopted [184]:

$$J = \frac{|C^1_{seq} \cap C^2_{seq}|}{|C^1_{seq} \cup C^2_{seq}|},$$

where C^1_{seq} = sequences that contain patterns from AP Cluster C^1 and C^2_{seq} = sequences that contain patterns from AP Cluster C^2.

The AP Cluster pairs are ranked by co-occurrence score and listed in descending order. When two or more AP Cluster pairs have the same score, the sequence count of the union of the two AP Clusters ($|C^1_{seq} \cup C^2_{seq}|$) is used as a secondary ranking criteria, i.e., the pair with a higher union size indicates that it covers more sequences and, hence, should be ranked higher.

Spectral clustering For spectral clustering [197], an adjacency matrix W is first created and filled with the co-occurrence score between the AP Clusters. Let W be an n by n matrix (n is the vertex count in G), where $W(i,j)$ is the adjacency weight between vertex v_i and v_j, i.e., the co-occurrence score between vertex v_i and v_j. The following matrices was first constructed:

$$d_i = \sum_j W(i,j).$$

$$D = diag(d_1, ..., d_n),$$

where D is an n by n matrix.

Next, using the adjacency matrix, a Laplacian matrix L is created, and L's eigenvectors are calculated. Using random walk, construct the Laplacian matrix

$$L_{rw} = I - D^{-1}W$$

where I is an n by n identity matrix. Find both the eigenvalues and their corresponding eigenvectors for L_{rw} and sort the eigenvectors by the ascending order of their eigenvalues.

Finally, the eigenvectors are then used as positions for the AP Cluster vertices v, with the weighted edges e being the Euclidean distance between v in the vertex space of G and its neighbours. K-means clustering is applied to G, minimizing the Euclidean distance of the eigenvectors between the vertices. Let k be the final cluster count, defined as the count before the largest difference between consecutive eigenvalues [197]. We use the first k columns in the eigenvectors for clustering. Each row in the eigenvector corresponds to an AP Cluster vector, with each vector having k values. Together the row and columns

make a point in k-dimensional space. Apply the k-means clustering algorithm on these given k points, but instead of maximizing similarities between the points within clusters, minimize the distances between the points.

Algorithm 8 Spectral clustering

Input: A set of AP Clusters $\mathbb{C}$, adjacency matrix W, and the final number of clusters required by the final k-means clustering algorithm
Output: AP Cluster Clusters $K_1...K_k$
for $i = 1$ to $|\mathbb{C}|$ **do**
  $d_i = \sum_j w_(i, j)$
end for
$D = diag(d_1, ..., d_n)$
Let I be a $|\mathbb{C}|x|\mathbb{C}|$ identity matrix
$L_{rw} = I - D^{-1}W$
Calculate the eigenvectors and their corresponding eigenvalues of L_{rw}
Sort the eigenvectors by their increasing eigenvalues
Take the first k columns of eigenvectors
Let each row of the eigenvector represent an AP Cluster,
and let each eigenvector column a dimension
Construct a k-dimension graph G_k with the eigenvector values
Apply k-means clustering on G_k, minimizing the Euclidean distance between the points within the clusters.
return $\{K_1...K_k\}$

Comparison of clustering algorithms Two other clustering algorithms are implemented to compare with spectral clustering: that is, the k-means clustering and the hierarchical clustering.

A special variation of the k-means clustering algorithm called k-medoids [23] is used in this paper. AP Clusters are used to represent the centroids since calculating a centroid with only co-occurrence scores between AP Clusters is difficult. The medoids are initialized

to be the first AP Cluster for each connected component due to the small number of AP Clusters considered. During the clustering process, the medoids are updated by finding the AP Cluster that maximizes the co-occurrence score between itself and all the other AP Clusters in the same cluster. Finally, to ensure that clustering provides the best possible results, five clustering indicators are computed to determine the optimal final number of clusters, i.e., optimum k, to be adopted for the k-medoids.

The hierarchical clustering algorithm uses a maximum spanning tree (MST) with minimal cut. First, an MST is built using Prims algorithm. Next, the minimal weighted edge of the MST is cut to separate the vertices, which are AP Clusters, into two co-occurrence clusters. The second step is repeated until an optimal solution is achieved.

The runtimess to find the optimal solutions for the three clustering algorithms are as follows: $O(n^4)$ for hierarchical clustering, $O(n^3)$ for spectral clustering, and $O(n^3)$ for k-medoids clustering. During the edge-cutting phase for hierarchical clustering the algorithm must evaluate all possible MST edges, a maximum of n edges, with each edge taking $O(n^2)$. Since there are a maximum of n MST edges to cut, the total running time is $O(n^4)$. K-medoids clustering takes $O(n^2)$ only if the cluster count is given. However, the algorithm is run n times to compare and obtain the optimal cluster count for the optimal clustering solution. Hence, the optimal solution has a runtime of $O(n^3)$. In comparison, spectral clustering takes $O(n^3)$ even with cluster count given, as the matrix multiplication that occurs when calculating the Laplacian matrix takes $O(n^3)$. However, the matrix is calculated only once, the optimal cluster count is obtained through the eigenvalues, and the algorithm uses the same that for the k-medoids algorithm to find the optimal cluster. Hence, the total runtime for spectral clustering is the same as k-medoids clustering,

$O(n^3)$. Because of the faster runtime, spectral and k-means clustering are preferred over hierarchical clustering.

Moreover, the spectral clustering algorithm is selected over the k-means clustering algorithm used in [110] because of the nature of the data. Pfam [63] sequences are built from multiple sequence alignments with the help of hidden Markov model; thus, the sequences have been pre-processed for correctness. UniProt [43] sequences are collected from a string query search of the database, so the quality of the sequences depends on the search terms. Therefore, the sequence quality of UniProt is less consistent, making it unsuitable for clustering using the global centroid of k-means since the low-quality sequences are heavily affected by outliers [86]. Closest neighbour characteristic in the spectral clustering algorithm is beneficial in handling noisy data. Therefore, this algorithm was selected for cluster co-occurring AP Clusters.

,

Verification by three-dimensional structure

To evaluate the importance of the AP Cluster regions discovered, we use the three-dimensional distance between the protein segments corresponding to the AP Clusters within the Co-occurrence Cluster. The rationale for using the three-dimensional distance is that if the AP Clusters are close together in three-dimensional space then they will likely interact with one another. It thus provides biophysical support that these functional regions are of biological importance to the proteins in the protein family tested.

After applying Co-occurrence Clustering, we manually select the cluster that contains

the lowest average eigenvector distance as the highly connected Co-occurrence Cluster. We relate these results to the corresponding three-dimensional protein structure from the Protein Data Bank (PDB) [20] using Chimera [147], highlighting the regions where the AP Clusters, or parts of the AP Clusters, appear. The distances between the AP Clusters are calculated as follows: the positions of each carbon alpha in each AP Cluster region is averaged, creating an average centroid for each AP Cluster region. The Euclidean distance is then calculated amongst all centroids. Finally, the AP Cluster distance is compared to the average pairwise distance, which is the average Euclidean distance of all possible carbon alpha pairs in the structure.

Using only the highly connected Co-occurrence Cluster and finding its biological importance, we validate 1) that the co-occurrence score ranks important AP Cluster pairs over the less important one, 2) that co-occurrence clustering is able to separate the less important AP Clusters out and 3) that our algorithm can provide reasonably good results in a timely manner, i.e. by not having to search through all AP Clusters discovered.

6.2.2 Datasets

The first dataset selected for our experiment contains two different protein families from UniProt, which are examined in subsequent detailed case studies. The first set is of ubiquitin protein sequences, downloaded on August 9th, 2012, with the following filters to obtain high quality sequences: having the name ubiquitin with a mnemonic starting with UB; and not containing the words ribosomal, modifier, factor, protein, conjugate, activating, or enzyme to remove other similar names. The second is of cytochrome c protein sequences,

downloaded on December 20, 2013, similarly with the filters: having the name cytochrome c with the mnemonic CY*; not ending in "ase" to prevent the inclusion of oxidase or reducatase; and not containing biogenesis or probability to remove other similar names. Each sequence from UniProt has an organism name, which is next searched in UniProt Taxonomy to acquire the condensed taxonomy lineage. Finally, the top kingdom name is extracted as the class label.

Next, our method was run on the two UniProt datasets. For the 70 ubiquitin input sequences, the pattern-discovery step was executed with a minimal length of 5, a maximum length of 15, a minimum occurrence of 20, and a delta of 0.9 (for control of delta closed pattern pruning). The maximum length restricted long (or high order) patterns from being discovered in the highly conserved ubiquitin sequences. Aligned pattern clustering was then executed with the following settings: Global Alignment with Hamming Distance and heuristics conditions with a minimum consecutive column match of 3, a minimum conserved column of 1, and no relative position overlapping. For the 319 cytochrome c input sequences, the pattern discovery step was executed with a minimal length of 5, a minimum occurrence of 40, and a delta of 0.9. The increase in the minimum occurrence was due to the increase in the number of input sequences. Aligned pattern clustering was then executed with the same settings as above. Lastly, the co-occurrence score was computed, and the three clustering algorithms were run. For both datasets, spectral clustering and k-medoids resulted in producing the same Co-occurrence Cluster.

The second dataset contains nine different protein families downloaded from Pfam Release 3.2 for a large-scale study of the three-dimensional structure of proteins. Pfam was used due to its well curated and pre-processed data. The proteins are lipocalin

[Pfam:PF000061]; bacterial rhodopsins [Pfam:PF00061]; bacterial antenna complex [Pfam:PF01036]; cytochrome c oxidase subunit I [Pfam:PF00115]; photosynthetic reaction centre protein family [Pfam:PF00124]; leptin [Pfam:PF02024]; G-alpha subunit [Pfam:PF00503]; protein kinase domain [Pfam:PF00069]; and tyrosine kinase [Pfam:PF07714]. The pattern-discovery and the aligned pattern clustering steps were executed with the same settings as above, except the minimum occurrence, which was adjusted based on the number of sequences and their sequence similarity as listed in Pfam. After clustering, we picked the Co-occurrence Cluster with the lowest average eigenvector distance to be evaluated for the three-dimensional distance.

6.3 Experimental results and discussions

6.3.1 Proteins verified by three-dimensional structure

We applied our method to nine protein families, confirming that our algorithm is effective at finding important regions on any protein family. Table 6.1 displays the Co-occurrence Cluster of closely related AP Clusters in the PDB structure of the related protein family. We found that these AP Clusters are close in Euclidean distance in the three-dimensional space.

Of interest are the results from the bacterial antenna complex family [Pfam:PF00556], where there is an average AP Cluster distance of 0 Å. The reason is that, despite having 5 AP Clusters in the maximum co-occurrence cluster, all AP Clusters overlap with one another, creating one long continuous region highlighted in blue (Figure 6.2). Furthermore,

Table 6.1: Results from the nine protein families. Displays the Co-occurrence Cluster with the lowest average eigenvector distance, and are used to verify the algorithm's effectiveness with a PDB structure. The shorter distance in the comparison is bolded. * means that one or more AP Clusters were not found.

Protein Name	Pfam ID	Co-occurrence Cluster Count	Size of the Best Cluster	PDB ID of the Best Cluster	Average AP Cluster Distance of the Best Cluster	Average Pairwise Distance
Lipocalin	PF00061	6	4	2CZT	**16.77 Å**	19.26 Å
Bacterial rhodopsins	PF01036	2	2	1JGJ	**16.52 Å**	22.51 Å
Bacterial antenna complex	PF00556	4	5	1IJD	**0 Å**	19.92 Å
Cytochrome c oxidase subunit I	PF00115	2	25	3OM3	**26.78 Å***	30.00 Å
Photosynthetic reaction centre protein family	PF00124	2	7	1PSS	**27.87 Å**	30.19 Å
Leptin	PF02024	2	14	1AX8	**15.73 Å**	18.37 Å
G-alpha subunit	PF00503	3	8	4G5O	**15.78 Å**	27.45 Å
Protein kinase domain	PF00069	2	2	3OZ6	**15.32 Å**	27.51 Å
Tyrosine kinase	PF07714	2	8	4HW7	**14.43 Å**	24.99 Å

the highlighted region covers positions 9 to 31 of the structure, and has only 46 amino acids, i.e., the maximum co-occurrence cluster continuously covers close to half of the whole structure. The figure also indicates that [Pfam:PF00556] might be highly conserved, exhibiting only minor variations in its primary structure across different proteins in the family, especially in the regions covered by the maximum co-occurrence cluster. Another result where the maximum co-occurrence cluster covers most of the amino acids in the PDB structure is Leptin [Pfam:PF02024, PDB:1AX8], where only 14 amino acids are not covered by the AP Clusters in the maximum co-occurrence cluster.

All the AP Clusters within the cluster in all the experiments in Table 6.1 were closer in distance than the average pairwise distance, indicating a relation between co-occurring AP Clusters and their distance in three-dimensional structures. We were able to observe some characteristics of the protein family, i.e., the conservation of its primary structure. Hence,

our algorithm is proven to discover important conserved regions for protein families.

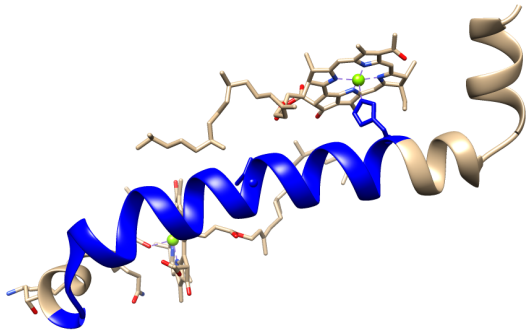


Figure 6.2: Three-dimensional structure of bacterial antenna complex [PDB:1IJD]. The set of all the patterns in the AP Clusters in the Co-occurrence Clusterinspected are all contained within one continuous highlighted blue region, indicating how the AP Clusters overlaps with one another.

6.3.2 Biological validation

In this section, we investigated the biological significance of Co-occurrence Clusters. Our experimental results revealed the Co-occurrence Clusters of ubiquitin and cytochrome c. Here we would like to study why co-occurring APCs are close to one another in spatial distance despite being far from each other in the primary sequence. Our hypothesis is that they need to form chemical bonding or co-operate in essential biological functions.

Ubiquitin case study

Ubiquitin (UBI) is a small (8.5kDa) protein that consists of a single polypeptide chain of 76 amino acids [196]. It plays an important role in ubiquitination, which is a post translational protein modification process where either a single ubiquitin or multiple chains of ubiquitin

are attached to a substrate protein. To form a chain, a ubiquitin connects to another ubiquitin by binding the diglycine in its C-terminal tail to one of the seven lysine amino acids of its linking partner.

Ubiquitination is widely used in regulating cellular signaling [53]. It does so by allowing the attached ubiquitin in substrate proteins to be bound through proteins with ubiquitin-binding domains (UBD) [53]. Either attaching a ubiquitin to a target protein or connecting it to another ubiquitin is regulated by the sequential activity of ubiquitin-activating (E1), ubiquitin-conjugating (E2) and ubiquitin-ligating (E3) enzymes [53].

When the seven lysine amino acids were mapped to our AP Clusters, they were all covered (Table 6.2). According to the results of our co-occurrence clustering algorithm in Figure 6.4, the optimum number of clusters of the six AP Clusters is two. The first cluster includes AP Cluster 1, 2, 3, 4 and 5; the second cluster includes AP Cluster 6 only. Their biological significance is discussed next.

The AP Clusters in the first cluster to co-occur for two reasons. First, each AP Cluster covers at least one Lysine (K). The diglycine in the C-terminal tail, i.e., Gly(G)75 and Gly(G)76 (green shade), is also covered in AP Cluster 3. As discussed earlier, Lysine (K) and the diglycine in the C-terminal tail are both important for the formation of multiple ubiquitin chains. Both AP Cluster 5 and AP Cluster 3 also cover important residues for facilitating the interaction of ubiquitin with E1 enzymes [34]. Mutagenesis experiments demonstrated that the mutation of Arg(R)42 or Arg(R)72 (red blocks) destabilizes the binding between Ubiquitin and E1 enzymes significantly, thus in turn, destroying the biological functions of ubiquitin [34]. Second, all AP Clusters except AP Cluster 5 cover the

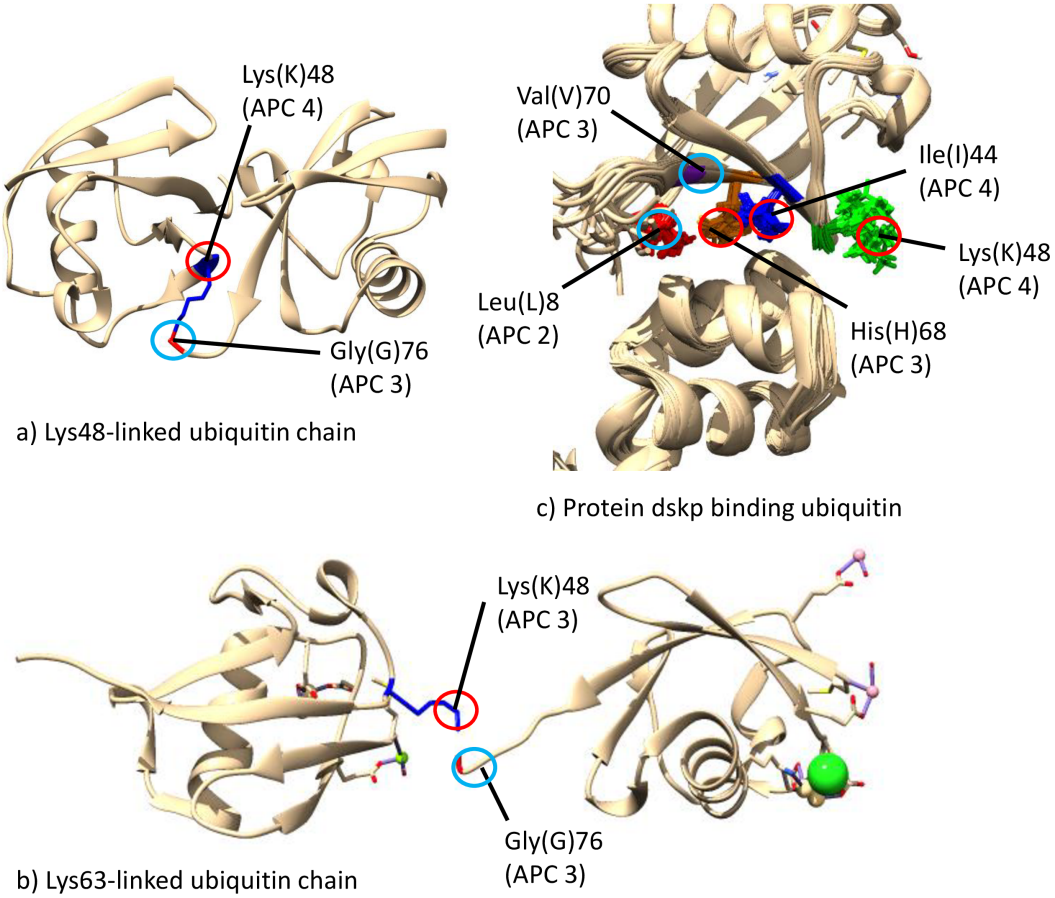


Figure 6.3: Three-dimensional structures of ubiquitin [PDB:1AAR,2JF5,1WR1]. The binding residues discussed in Table 6.2 and their functions are displayed. a) is the ubiquitin chain linked by the Lys(K)48 in APC 4 to the diglycine, b) is the ubiquitin chain linked by the Lys(K)63 in APC 4 to the diglycine, c) is the binding between dskp binding ubiquitin and ubiquitin by Leu(L)8 of APC 2, Val(V)70 of APC 3, Ile44(I) and Lys(K)48 of APC 4, and His(H)68 of APC 3.

Table 6.2: Key residues covered by AP Cluster and their roles in the Co-occurrence Cluster 1 of ubiquitin

AP Cluster	Residue(s)	Role(s)	Literature
1	K6, K11	Lys(K)6 and Lys(K)11 are used for forming ubiquitin chain(s) in ubiquitination.	[196]
	L8	Leu(8) facilitates the interaction between ubiquitin and E1 enzymes.	[53, 185]
2	K11, K27	Lys(K)11 and Lys(K)27 are used for forming ubiquitin chain(s) in ubiquitination.	[196]
	L8	Leu(8) facilitates the interaction between ubiquitin and E1 enzymes.	[53, 185]
3	K63	Lys(K)63 is used for forming ubiquitin chain(s) in ubiquitination.	
	H68, V70	His(H)68 and Val(V)70 facilitate the binding between ubiquitin and ubiquitin-binding proteins.	[53, 185]
	R72	Arg(R)72 facilitates the interaction between ubiquitin and E1 enzymes.	[34]
	G75,G76	Gly(G)75 and Gly(G)76 are are used for forming ubiquitin chain(s) in ubiquitination.	[196]
4	R42	Arg(R)42 facilitates the interaction between ubiquitin and E1 enzymes.	[34]
	I44	Ile(I) 44 is the binding site between ubiquitin and the ubiquitin-binding proteins.	[53, 185]
	K48	Lys(K)48 is used for forming ubiquitin chain(s) in ubiquitination. It also facilitates the binding between ubiquitin and ubiquitin-binding proteins.	[196, 53, 185]
5	K27,K29,K33	Lys(K)27, Lys(K)29 and Lys(K)33 are used for forming ubiquitin chain(s) in ubiquitination.	[196]
	R42	Arg(R)42 facilitates the interaction between ubiquitin and E1 enzymes.	[34]

ubiquitin-binding residues. These residues are important for the tight binding of ubiquitin with ubiquitin-binding proteins [53]. Therefore, the AP Clusters in the Co-occurrence Cluster 1 are due to both ubiquitination and ubiquitin-binding.

There is only one AP Cluster, AP Cluster 6, in the second cluster (Figure 6.4) which has no co-occurrence with other AP Clusters. We also observed a certain degree of overlapping

between AP Cluster 6 and AP Cluster 5. We propose two reasons to explain why AP Cluster 6 is not merged with AP Cluster 5 but exists alone in another cluster. First, the conserved amino acid in residue 24 of AP Cluster 6 and AP Cluster 5 is Asp(D)24 and Glu(E)24 (yellow shade), respectively. We found that ubiquitin of Viridiplantae (plant kingdom) has mostly Glu(E)24, whereas ubiquitin of Metezoa (animal kingdom) has mostly Asp(D)24 in our dataset, this site is also well-known for differentiating human (containing Glu(E)24) ubiquitin from yeast (containing Asp(D)24) ubiquitin [195]. Hence, AP Cluster 6 and AP Cluster 5 are not merged in this study, because they cover patterns with different amino acids in different species.

Second, AP Cluster 6 does not include ubiquitination-related Arg(R)42 and covers the alpha helix 1, from residues 23 to 34, more precisely than AP Cluster 5. Previous literature has discovered that alpha helix 1 is an unconventional recognition site of ubiquitin-binding proteins [185]. Experiments in the same study revealed that, even if Ile(I)44 and His(H)68 were mutated, a high affinity binding between protein CKS1 and ubiquitin would still be identified, thereby proving that ubiquitin is unconventionally bound by CKS1 [185]. It should be noted that the conventional and unconventional ubiquitin-binding is not mutually exclusive [185]. Hence, AP Cluster 5 in the first cluster and AP Cluster 6 in the second cluster are not merged. Where AP Cluster 5 represents the scenario that either only conventional ubiquitin-binding occurs or conventional and unconventional ubiquitin-binding co-occur, AP Cluster 6 represents the scenario that only unconventional ubiquitin-binding occurs. Our experimental results from ubiquitin and literature search give us very strong support for the biological significance of the discovered Co-occurrence Cluster.

Cytochrome c case study

Cytochrome c (cyt-c) is a small (12.4kDa), heme-containing protein that consists of approximately 104 amino acids [226]. It is an essential component of the electron transport chain in the mitochondria. The heme group of cyt-c accepts electrons from the complexes III (cytochrome b-c_1 complex or cyt-bc_1) and transfers electrons to the complexes IV (cytochrome c oxidase or cyt-c_1) [226].

According to the results of our co-occurrence clustering algorithm (Figure 6.6), the optimum number of clusters of the 8 AP Clusters is 2. The first cluster includes AP Clusters 1 to 3; the second cluster includes AP Clusters 4 to 8. Their biological significance is discussed as below.

For the first cluster, we found that all the AP Clusters covered residues that contributed significantly to the binding of cyc-1 on cyc-bc_1. This is crucial for electron transfer. Experiments have established the importance of Lys(K)8, Lys(K)27 and, to a lesser extent, Lys(K)5, Lys(K)7, Lys(K)25 [182, 161, 97]. They are covered in the AP Clusters in the first cluster (Table 6.3). Therefore, these AP Clusters co-occur to facilitate the binding of cyc-1 on cyc-bc_1.

Table 6.3: Key residues covered by AP Clusters and their roles in co-occurrence cluster 1 of cytochrome c

AP Cluster	Residue(s)	Role(s)	Literature
1	Lys25, Lys27	The binding sites of cytochrome c cytochrome BC_1 complex	[182, 161, 97]
2	Lys27	The binding sites of cytochrome c cytochrome BC_1 complex	[182, 161, 97]
3	Lys5, Lys7, Lys8	The binding sites of cytochrome c cytochrome BC_1 complex	[182, 161, 97]

For the second cluster, we found that all the AP Clusters covered residues that were mostly responsible for the stable axial ligand between cyc-t and the heme group (Figure 6.5), which is the component that takes part in the redox reactions for the electron transfer between cyt-c and other complexes. AP Cluster 4 covered Cys(C)14 [19, 29], Cys(C)17 [19, 29] and His(H)18 [74, 183]. His(H)18 [74, 183] forms an axial ligand with the heme from the proximal front. Cys(C)14 [19, 29] and Cys(C)17 [19, 29] enhance and maintain the axial ligand between His18 and the heme. AP Cluster 5 covered Tyr(Y)67 [200, 226], Pro(P)71 [199], and Pro(P)76 [24], Met(M)80 [183] and Phe(F)82 [127]. Met(M)80 [183] forms an axial ligand with the heme from the distal side. Tyr(Y)67 [200, 226], Pro(P)71 [199], Pro(P)76 [24] stabilize and coordinate the axial ligand between Met(M)80 and the heme. Phe(F)82 [127] stabilizes the native heme environment. AP Cluster 6 covered Gly(G)41 [89], which holds the axial ligand between Met(M)80 and the heme. AP Cluster 7 covered Asn(N)52 [163, 166], which maintains a hydrogen bond with the heme to stabilize the environment.

Although AP Cluster 8 did not cover any residues that are directly related to the axial ligands between cyt-c and the heme group, it covered residues that maintain the cyt-c structure. Among the 38 intra-molecular hydrophobic interactions reported in [163], AP Cluster 8 covered 17 (44.7%). It also covered Leu(L)94 [68] and Tyr(Y)97 [68], where one of them is required to provide a hydrophobic environment in order for cyt-c to function. Evidently, the AP Clusters in the co-occurrence cluster 2 form and maintain stable axial ligands with the heme and also provide an appropriate structure and environment for cyt-c to function.

Table 6.4: Key residues covered by AP Clusters and their roles in co-occurrence cluster 2 of cytochrome c

AP Cluster	Residue(s)	Role(s)	Literature
4	Cys(C)14	Cys(C) 14 enhances axial ligand strength between His18 and the heme.	[19, 29]
	Cys(C)17	Cys(C) 17 enhances axial ligand strength between His18 and the heme.	[19, 29]
	His(H)18	His(H)18 forms an axial ligand with the heme from the proximal front.	[74, 183]
5	Tyr(Y)67	Tyr(Y)67, its hydroxyl group, forms a H-bond with side chains of Met80 for structural stabilization.	[200, 226]
	Pro(P)71	Pro(P)71 helps coordinate the axial ligand between Met80 and the heme.	[199]
	Pro(P)76	Pro(P)76 helps coordinate the axial ligand between Met80 and the heme.	[24]
	Met(M)80	Met(M)80 forms an axial ligand with the heme from the distal side.	[74, 183]
	Phe(F)82	Phe(F)82 helps stabilize the native heme environment.	[127]
6	Gly(G)41	Gly(G)41 helps stabilize the axial ligand between Met80 and the heme.	[89]
7	Asn(N)52	Asn(N)52 maintains a hydrogen bond with the heme to stabilize the environment.	[163, 166]
8	Leu(L)94	One of Leu(L)94 or Tyr(Y)97 is required to provide a hydrophobic environment for the function of cyt-c.	[68]
	Tyr(Y)97	One of Leu(L)94 or Tyr(Y)97 is required to provide a hydrophobic environment for the function of cyt-c.	[68]

6.4 Conclusion

In this dissertation chapter , we address the two research questions that were first posed in the introduction. We answer the first research question on discovering co-occurrences by creating a novel algorithm that clusters AP Clusters with frequently co-occurring patterns into an effective, statistical, and measurable Co-occurrence Clusters. We respond to the second research question on the biological significance of these Co-occurrence Clusters by

their three-dimensional closeness and by biological functionality and structural integrity. We confirm that the Co-occurrence Cluster with the lowest average co-occurrence score is also closer in three-dimensional distance than the average amino acids in the three-dimensional structure. We also confirm that co-occurring AP Clusters form chemical bonds or co-operate in essential biological functions as supported in biological literature. As a natural extension, we can use correlated amino acid variations to track evolutionary divergence and extend the algorithms to discover consistence and deviance of chemical properties. Since it is time-consuming to study the functional and structural sites for every target protein's drug interaction in detail, the ability to discover top-ranking Co-occurrence Clusters could also help to isolate the amino acids of biological significance. Hence, our method will have great potential to impact drug discovery and the biomedical community.

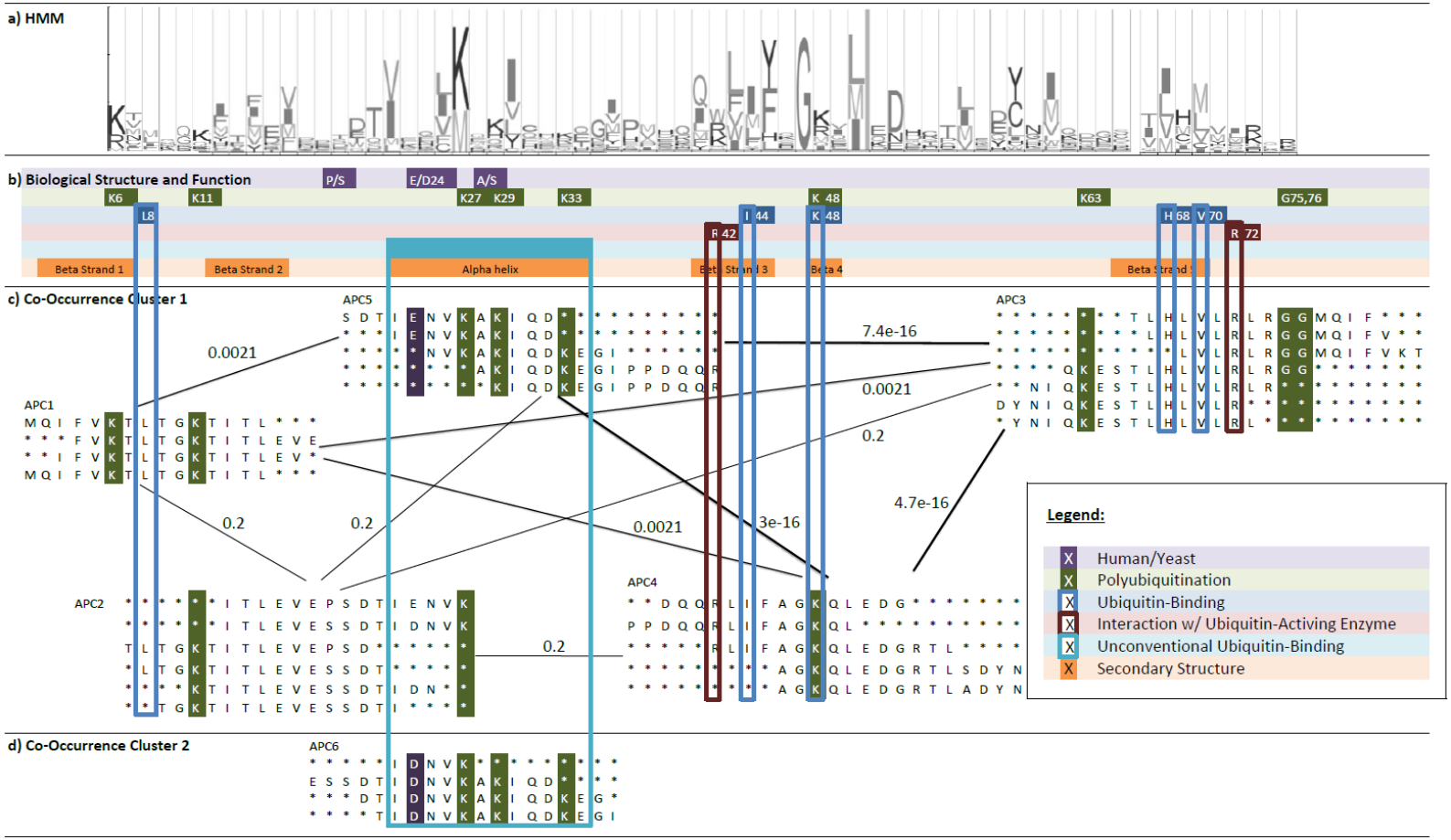


Figure 6.4: Co-occurrence clusters of ubiquitin.General Features: a) the top of the diagram is part of the HMM sequence profile of ubiquitin; b) the color shading blocks with legends immediately below mark the important amino acids and segments forming the important structure and function of the protein; c-d) the APCs discovered are represented by arrays of aligned amino acids; the color shaded columns correspond to the significant residues marked as in b); if the co-occurrences of patterns between APCs are frequent, the co-occurrence APCs are linked by an edge with weight representing co-occurrence score; treating APCs as vertices. A co-occurrence APC cluster is represented by a weighted graph linking co-occurring APC s; the important functional regions of the molecules as listed in Table 6.2 are highlighted in colored blocks specified by the legend. Specific Features: Note that APC 5 and APC 6 are not linked by co-occurrence since they belong to different taxonomical group and with different amino acids, Asp(D)24 and Glu(E)24, in the same column.

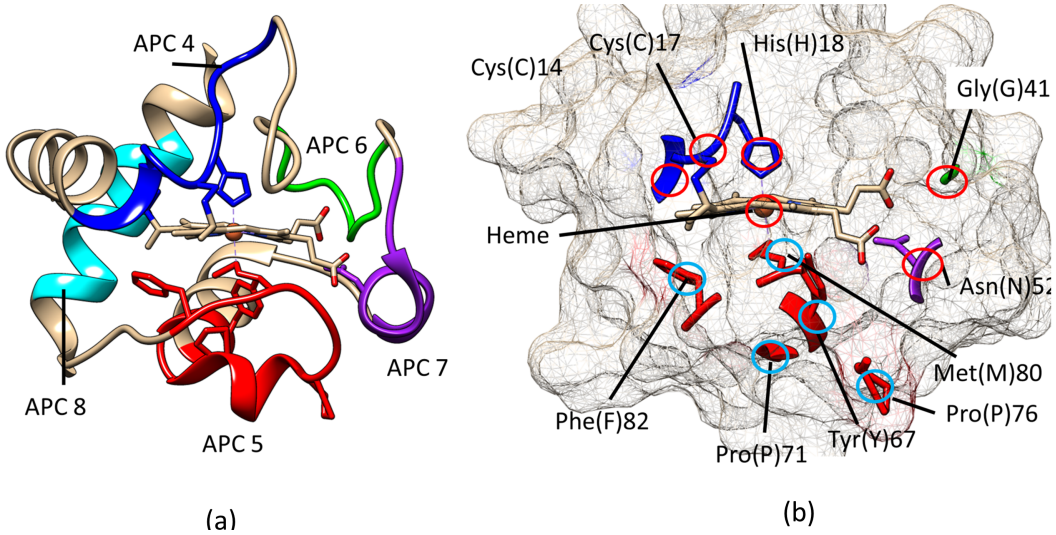


Figure 6.5: Three-dimensional structure of cytochrome c [PDB:1HRC].a) The APCs in Co-occurrence Cluster2 as listed in Table 6.4. b) The amino acids from APCs in Co-occurrence Cluster2 mostly interact with the heme to stabilize the axial ligand, as confirmed by biological literature listed in Table 6.4.

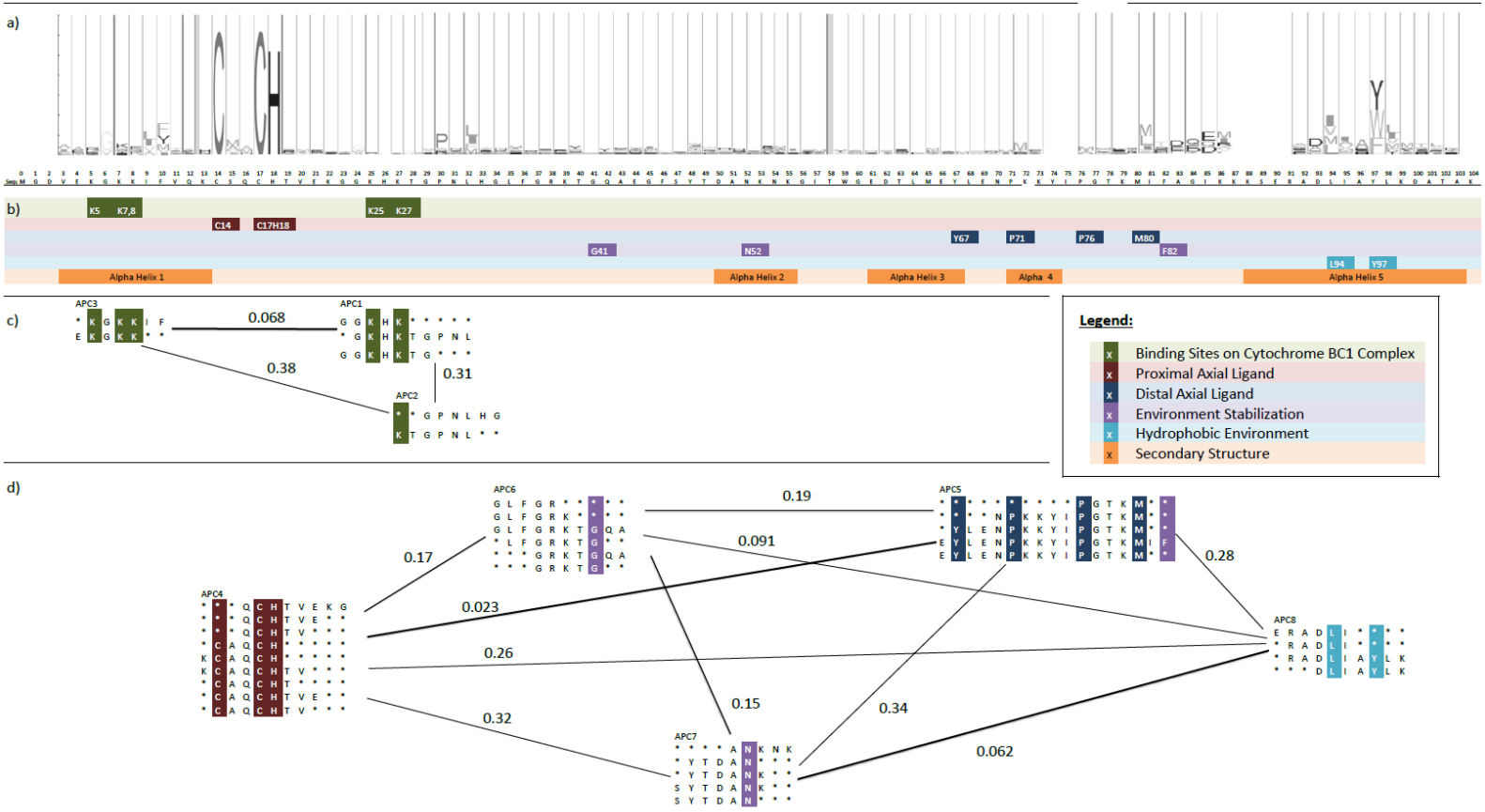


Figure 6.6: Co-occurrence clusters of cytochrome c.General Features is same as stated in Figure 6.4 c-d) Important functional regions as listed in Tables 6.3 and 6.4, are highlighted here in color blocks as specified by the legend; Specific Features: Amino acids in Co-occurrence Cluster 1 facilitate the binding of cyc-1 on cyc-bc1 as listed in Table 6.3 and most of the amino acids in Co-Occurrence Cluster 2 are responsible for the stable axial ligand between cyc-t and the heme group.

Chapter 7

Conclusion

> It is not the strongest of the species that survives, nor the most intelligent that survives. It is the one that is the most adaptable to change.
>
> Charles Darwin

7.1 Concluding Remarks

The focus of this dissertation is clustering sequence patterns, which are biologically validated as meaningful and useful. These stable patterns (motifs) with variations are related to known protein binding functions (Chapter 3 and 4), are accurately and efficiently revealing class characteristics (Chapter 5), and are structurally and functionally significant if co-occurring on the same protein (Chapter 6). The experimental case studies from *in*

silico biological datasets are qualitatively confirmed with statistical significance to reflect regional functionality such as binding segments and sites. The class characteristics are confirmed through six proposed cluster validity measures using external class labels or using only the information inherent in the data alone in revealing the commonalities and differences in the class types. Finally, the significance of co-occurrence of AP Clusters is supported biologically by protein three-dimensional structure and through amino acid interactions between distant regions.

Methods in the literature for biological pattern analysis include database search, multiple sequence alignments, motif finding, traditional classification, and other co-occurrence techniques. Current methods are insufficient at identifying the functional regions of proteins, although some solutions have been proposed. First, protein databases containing existing experimental results and annotations do not contribute additional new knowledge to the molecule. Sequence alignment approaches such as global alignment and local alignment also have their short-cominings. Global alignment of sequences (multiple sequence alignment) does not perform well in sequences with low similarity caused by evolutionary divergence and the parameters of local pattern discovery (motif finding) require greater flexibility in fixed parameters such as length, number of variations, and number of degeneracies. Until the knowledge discovered in the sequences is organized, external class labels and pattern interactions cannot be used to learn new knowledge. Existing supervised classification are inherently biased by mislabelling, incorrect partitioning, and unbalanced classes. Existing methods for discovering co-occurrence between patterns include evolutionary tracing which requires whole-sequence similarity and coupling analysis which requires a large number of homologuous non-redundant sequences.

To overcome the hurdles as summarized above, Aligned Pattern Clusters (AP Clusters) were developed in this dissertation to represent compact yet rich patterns with variations especially when prior knowledge is unavailable. They capture conservations and variations by covering more sequences with lower entropy with a greatly reduced number of patterns. They contain statistically significance patterns with variations and their importance has been confirmed by the following biological evidence: 1) Most of the discovered AP Clusters correspond to binding segments while their aligned columns correspond to binding sites as validated by pFam, PROSITE, and the three-dimensional structure. 2) By compacting strong correlated functional information together, AP Clusters are able to reveal class characteristics for taxonomical classes, gene classes and other functional classes through simple cluster validity measures unaffected by mislabelling biases, incorrect partitioning, unbalanced classes, or unknown functional classes. 3) Co-occurrence of AP Clusters on the same homologous protein sequences are spatially close in the protein's three-dimensional structure and their interacting amino acids are functionally important. These results demonstrate the power and usefulness of AP Clusters. They bring in similar statistically significant patterns with variation together and align them to reveal protein regional functionality, class characteristics, binding and interacting sites for the study of protein-protein and protein-drug interactions for cancer tumours differentiation, targeted gene therapy as well as drug target discovery.

7.2 Future Work on Drug Discovery and Next Generation Sequencing

The next wave of major advances in life science and healthcare will be spurred by the advent of bioinformatics and computing power, in which next-generation sequencing (NGS) technology and drug-discovery will play an important role. The main obstacle for NGS is not in generation of data but in their analysis. The use of AP Clusters in discovering new knowledge using a data-driven approach will be crucial for targeting the regions and amino acids of critical biological significance. Already, metagenomics researchers are studying microbacterial biodomes similar to traditional microarray techniques and cancer researchers are classifying cancer tumours using a traditional classification approaches such as random forests. The contributions of this thesis in discovering compact yet rich patterns with variations can be applied in identifying signals in next-generation sequencing (NGS), as well as differentiating the single nucleotide polymorphisms (SNPs) of cancer tumour types. With NGS, we are able to sequence patients who are sensitive to a drug and other patients who are sensitive to another drug. With appropriate software tools, we can discover the patterns within the different groups. These patterns will help us to recommend the best drug for patients with a disease like cancer. This is what we call Personalized Medicine. Although it seems that we still have a long way to go, we are moving very fast towards the goal. In addition, the co-occurrence contribution of this dissertation is currently proven for distant within-protein interaction and being further developed for protein-DNA binding, protein-protein interaction, and protein-drug interaction.

Appendix A

Biological Background

This background chapter is designed to include only the minimal biological background required to understand the premise of this dissertation. The first section begins broadly with a principle which applies to all organisms: the central dogma of molecular biology. The protein is described in terms of its static structural organization by breaking down the level of protein structure organization, i.e. a hierarchical abstraction building protein from primary to secondary to tertiary to quaternary structures.

A.1 Protein Biochemistry

A.1.1 The Central Dogma of Molecular Biology

The three biological sequences that encode the functions of life are DNA, RNA, and protein. Today, the full human genome sequences has been decoded, but decoded remains

disorganized. The challenge is to determine the protein structure that demonstrates the function behind this genomic sequence information. It is the final product of the protein sequence, a protein structure, that enables the complex functions within the cell.

The Central dogma of molecular biology demonstrate how information from four letters alphabet, or four different nucleotides, of DNA flows to RNA and turns into the twenty letter alphabet, or twenty different amino acids, of protein. The dogma is central because its processes occur in all living cells. The central dogma of molecular biology describes how DNA, RNA, and protein form from one to the other: DNA either replicates itself or is transcribed into RNA intermediary molecule; RNA groups into three letter code, or codon, and translates to protein [2]. The focus of this thesis is on protein structure prediction, thus the next section will focus on the third macromolecule, the protein.

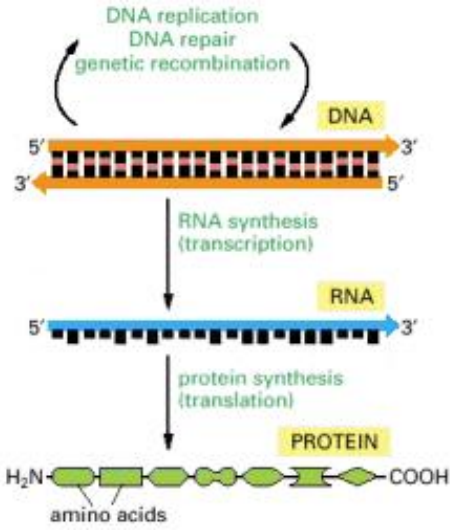


Figure A.1: Central dogma of molecular biology describe the flow of genetic information from DNA to RNA (transcription) and from RNA to protein (translation).

A.1.2 Levels of Protein Structure Organization

To introduce the protein, this section considers the static components of protein structure while the next section consider the dynamic aspect of protein folding. Protein can be divided into structural subunits: from the fundamental building block of amino acid, to the primary sequence, all the way to the quaternary structure. The energy of protein folding forces it into its structure, this energy is important to the abstraction of structural classification.

Amino Acid as the Fundamental Building Block of Protein

Amino acids are the smallest building blocks that assemble together to create proteins. It is defined by one carboxylic acid group and one amino group; thus it is called amino from the amino group and acid from the carboxylic acid group. The amino acid is anchored by a central alpha-carbon, which connects the amino group and the carboxylic acid group, in addition to a hydrogen and a variable side-chain (R).

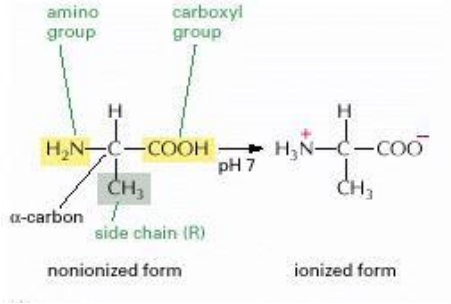


Figure A.2: Chemical Formula of an Amino Acid: The chemical formula of an amino acid, which consists of a central alpha-carbon connected by a amino group, a carboxylic acid group, a hydrogen and a variable side-chain(R).

Twenty different possible side-chains can attach to the central alpha-carbon atom of the amino acids. Each possible amino acid has its own set of distinct properties, such as hydrophobic or hydrophilic, charge or uncharged, acid or base, bulkiness, and many others.

Polypeptide Chain as the Primary Structure

The first level of protein structure organization is the primary structure, which is a linear sequence of amino acids chained together into a polypeptide chain. Two amino acid are joined by a peptide bond and when multiple amino acids are joined head-to-tail into a long chain, a polypeptide is created. Along the core of peptide chain is the polypeptide backbone

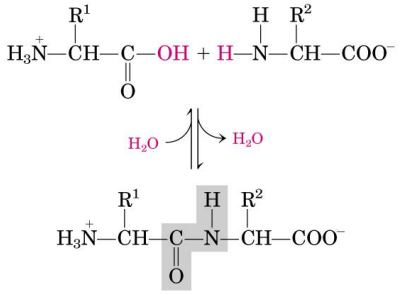


Figure A.3: Peptide Bond: Two amino acids react with one another to give off one water and forms one peptide bond.

consists of repeating sequence of carbon and nitrogen atoms. A polypeptide has definite direction with endings: the amino end (NH2)of polypeptide is called the N-terminus, and the carboxyl (COOH) end of the polypeptide is called the C-terminus.

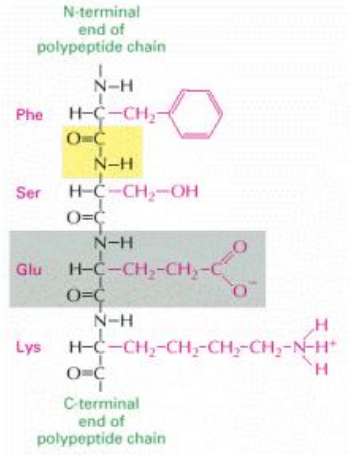


Figure A.4: Polypeptide Chain: Multiple amino acids are chained together by multiple peptide bonds to form a polypeptide chain.

Hydrogen Bonds Form Regular Substructures as Secondary Structure

The polypeptide chain forms the primary structure, which interacts with its own three-dimensional space to form substructures. The hydrogen bond is an interaction between the N-H group of one amino acid and the C=O group in another amino acid, both amino acids are from polypeptide backbone of the chain. Because it does not involve the variability of side-chain characteristics, hydrogen bond is a widely common interaction without needing specificity the exact side-chain. A regular repeating conformation of these hydrogen bonds form two regular fold patterns: the alpha-helix and the beta-sheet.

The three secondary structures are: alpha helix, beta-sheet, and loop. First, the alpha helix appears like a twisted telephone cord with regular hydrogen-bond between every first and fourth residue. In this way, the i^{th} amino acid forms a hydrogen bond locally with the $i{+}4^{th}$ amino acid. The alpha helix is a simple regular structure which forms a complete turn every 3.6 amino acid. The beta sheet looks like a sheet where segments of the polypeptide

chain line up next to one other and form hydrogen bonds. Its hydrogen bonds are between two distant strands of the polypeptide chain running side by side. If these two strands are going in the same direction, then the beta-sheet is called parallel. If one strand folds back on itself on the second strand causing them to go the opposite direction, then the beta-sheet is called anti-parallel. Compared to the simple local hydrogen bonds in alpha helix, the distance between beta-strands causes the prediction difficulty. Finally, a loop has no definite structure, and usually links other secondary structures.

Tertiary and Quaternary Structures as Higher Structural Organizations

A tertiary structure is a polypeptide chain formed by secondary structures assembled into a full three-dimensional structure. Quaternary Structure is a complex protein built from subunits of multiple tertiary structures which are folded polypeptide chains.

Appendix B

Terminology

B.1 Glossary of Terms

An Alphabet: is a collection of symbols that may be augmented with '', which acts as an empty symbol. For convenience, denote $\Sigma = \hat{\Sigma} \cup -$, where $\hat{\Sigma} = \{\sigma_1, \sigma_2, \ldots, \sigma_{|\Sigma|-1}, \sigma_{|\Sigma|}\}$

A Sequence: $\mathbb{S} = \{s^k | k = 1, ..., |\mathbb{S}|\} = \{s^1, s^2, \ldots, s^{|\mathbb{S}|-1}, s^{|\mathbb{S}|}\}$

A Set of Unaligned Patterns: $\bar{\mathbb{P}} = \{\bar{p}^i | i = 1, ..., |\bar{\mathbb{P}}|\} = \{\bar{p}^1, \bar{p}^2, \ldots, \bar{p}^{|\bar{\mathbb{P}}|-1}, \bar{p}^{|\bar{\mathbb{P}}|}\}$

An Unaligned Pattern: $\bar{p}^i = s^i_1 s^i_2 ... s^i_{|\bar{p}^i|}$

A Set of Aligned Patterns: $\mathbb{P} = \{p^i | i = 1, ..., |\mathbb{P}|\} = \{p^1, p^2, \ldots, p^{|\mathbb{P}|-1}, p^{|\mathbb{P}|}\}$

The Occurrence of the Pattern $\bar{p}^i$: $occ(\bar{p}^i) = j_i$ such that $\bar{p}^i = s^i_{j_i} s^i_{j_i+1} \dots s^i_{j_i+|\bar{p}^i|-1}$, where i is the index of the sequence that pattern occurs in, and j_i is the starting index the pattern in that sequence.

A Set of AP Clusters: $\mathbb{C} = \{C^l | l = 1, ..., |\mathbb{C}|\} = \{C^1, C^2, \dots, C^{|\mathbb{C}|-1}, C^{|\mathbb{C}|}\}$

An AP Cluster:

$$C^l = \texttt{ALIGN}(\mathbb{P}^l), \tag{B.1}$$

$$= \begin{pmatrix} s^1_1 & s^1_2 & \dots & s^1_n \\ s^2_1 & s^2_2 & \dots & s^2_n \\ \vdots & \vdots & \vdots & \vdots \\ s^m_1 & s^m_2 & \dots & s^m_n \end{pmatrix}_{m \times n} = \begin{pmatrix} p^1 \\ p^2 \\ \vdots \\ p^m \end{pmatrix}, \tag{B.2}$$

$$= \begin{pmatrix} c_1 & c_2 & \dots & c_n \end{pmatrix}. \tag{B.3}$$

where $s^i_j \in \Sigma \cup \{-\} \cup \{*\}$ ispattern p^i with a newly aligned column index j. Each of the $|\mathbb{P}^l| = m$ patterns in the rows of C^l is of length $|C^l| = n$.

An Aligned Pattern: $p^i = s^i_1 s^i_2 ... s^i_{|p_i|}$ is a subsequence of order-preserving elements maximizing the similarity of the patterns against a set of patterns from AP Cluster, $\mathbb{P}_l$, with gaps, wildcards, and mismatches to the length $|\mathbb{P}^l| = n$.

An Aligned Column: Let c_j in C^l represents the j^{th} column of amino acids from the set of patterns that forms the current AP Cluster, $C^l = \big(c_1, c_2, \dots, _\backslash\big)$.

Distinct amino acids in the aligned column c_j: $\Sigma(c_j) = \{s_j^i = \sigma | p^i = s_1^i ... s_j^i ... s_n^i, p^i \in \mathbb{P}^l, \sigma \in \Sigma \cup \{-\} \cup \{*\}\}$. We denote $\sigma(c_j)$ as an amino acid in $\Sigma(c_j)$.

Data Induced by the Unaligned Pattern: Let $\mathbb{D}(\bar{p}^i)$, be all the occurrences of the pattern, $\bar{p}^i$, that is in the input sequence. We call $\mathbb{D}(\bar{p}^i)$ the data induced by $\bar{p}^i$ or the induced data of $\bar{p}^i$. We will return to the concept for AP Cluster which is later used for computing the measures for aligned columns.

Data Induced by AP Cluster: Let $\mathbb{D}(C^l)$ be data induced by the AP Cluster C^l, which is the subset of segments from the input sequences, or the data subspace containing all the pattern from the AP Cluster, C^l, $\mathbb{P}^l = \{p^1, p^2, \ldots, p^m\}^T$. We call $\mathbb{D}(C^l)$ the data induced by C^l or the induced data of C^l. Then $\mathbb{D}(C^l)$ is then the union of the segements from the input sequences induced by all the patterns contained in C^l, $\mathbb{D}(C^l) = \mathbb{D}(p^1) \cup \mathbb{D}(p^2) \cup \cdots \cup \mathbb{D}(p^m) = \bigcup_{\forall p^i \in \mathbb{P}^l} \mathbb{D}(p^i)$

A AP Hypergraph: An Aligned Pattern Directed Hypergraph (AP Hypergraph) is a directed graph, $G = (\mathbb{V}, \mathbb{E})$, where vertices and directed edges are defined as follows:
$\mathbb{V} = \{\nu_j(\sigma) | 1 \leq j \leq n, \sigma \in \Sigma, \mathbb{P}(\nu_j(\sigma)) \neq \emptyset\}$, where $\mathbb{P}(\nu_j(\sigma)) = \{P \in \mathbb{P} | s^j = \sigma\}$
$\mathbb{E} = \{\epsilon_j(\nu_j(\sigma), \nu_{j+1}(\sigma')) | 1 \leq j \leq n, \sigma, \sigma' \in \Sigma, \mathbb{P}(\nu_j(\sigma)) \bigcap \mathbb{P}(\nu_{j+1}(\sigma')) \neq \emptyset\}$

Data Induced by AP Hypergraph: Let $\mathbb{D}(G^l) = \mathbb{D}(C^l)$ be data induced by the AP Hypergraph G^l, which is the subset of segments from the input sequences, or the data subspace containing all the pattern from the AP Hypergraph, G^l, $\mathbb{P}^l = \{p^1, p^2, \ldots, p^m\}^T$.

We call $\mathbb{D}(G^l)$ the data induced by G^l or the induced data of G^l. Then $\mathbb{D}(G^l)$ is then the union of the segements from the input sequences induced by all the patterns contained in G^l, $\mathbb{D}(G^l) = \mathbb{D}(p^1) \cup \mathbb{D}(p^2) \cup \cdots \cup \mathbb{D}(p^m) = \bigcup_{\forall p^i \in \mathbb{P}^l} \mathbb{D}(p^i)$

A pattern hyperedge: In the context of the above notations, the i^{th} pattern is $\mathbb{V}^i = \{\nu_1(\sigma_1), \nu_2(\sigma_2), ..., \nu_n(\sigma_n) | \sigma_1\sigma_2...\sigma_n = P_i\}$.

A column hyperedge: In the context of the above notations, the j^{th} aligned column is, $\mathbb{V}^j = \{\mathbb{P}(\nu_j(\sigma)) | \sigma \in \Sigma, \mathbb{P}(\nu_j(\sigma)) \neq \emptyset\}$

An association hyperedge: The interdependency is between the data represented by the vertices of the two column hyperedge.

Coverage: Total input sequences that are covered by the AP Hypergraph.

AP Hypergraph Quality: $Q = 1 - \frac{1}{n} \sum_{j=1}^{n} H(c_j)$.

Standard Residual: $\texttt{StandardResidual} = \frac{o-e}{\sqrt{e}}$ where $e = N\left(\prod_{i=1}^{n}\left(\sum_{\forall \sigma_k \in \mathbb{V}^j} Pr(\nu_j(\sigma_k))\right)\right)$.

Co-occurrence Score (The Jaccard index): $J = \frac{|C^1_{seq} \cap C^2_{seq}|}{|C^1_{seq} \cup C^2_{seq}|}$ where C^1_{seq} = sequences that contain patterns from AP Cluster C^1 and C^2_{seq} = sequences that contain patterns from AP Cluster C^2

Table B.1: Four Cluster Validity Measures for column hyperedges

	External Measure using External Class Labels	Internal Measure using Data Alone(Summed and Normalized)
1.	External Entropy Normalized Class Information Entropy $H_Y(R) = -\frac{1}{log(\lvert Y\rvert)}\big(\sum_{y_i \in Y} pr(y_i) log(pr(y_i))\big)$.	Internal Entropy Normalized Amino Acid Information Entropy $H(c_i) = H = -\frac{1}{log(\lvert\Sigma(c_i)\rvert)}\big(\sum_{\sigma(c_i)\in\Sigma(c_i)} pr(\sigma(c_i)) log(pr(\sigma(c_i)))\big)$. $R1 = 1 - H(c_i)$
2.	External Information Gain Class Information Gain $\Delta H_Y(c_j) = \frac{1}{H_Y(c_j)}\Big(H_Y(c_j) - \sum_{\sigma(c_j)\in\Sigma(c_j)} \big(w(\sigma(c_j)) H_Y(\sigma(c_j))\big)\Big)$.	Internal Information Gain Normalized Sum of Information Gain $IG = \Delta H(c_i\lvert c_j) = \frac{1}{H(c_i\lvert c_j)}\Big(H(c_i\lvert c_j) - \sum_{\sigma(c_j)\in\Sigma(c_j)} \big(w(\sigma(c_j)) H(c_i\lvert\sigma(c_j))\big)\Big)$, $SIG(c_i) = \frac{1}{n}\sum_{j=1}^{n} IG(c_i, c_j)$.
3.	Information Correlation	
a.	External Joint Entropy Joint Entropy with Class $H(c_i, Y) = -\sum_{\sigma(c_i)\in\Sigma(c_i)} \big(\sum_{y_i\in Y} pr(\sigma(c_i), y_i) log_2(pr(\sigma(c_i), y_i))\big)$.	Internal Joint Entropy Joint Entropy by Amino Acid $H(c_i, c_j) = \sum_{\sigma(c_i)\in\Sigma(c_i)} \big(\sum_{\sigma(c_j)\in\Sigma(c_j)} pr(\sigma(c_i), \sigma(c_j)) log_2(pr(\sigma(c_i), \sigma(c_j)))\big)$.
b.	External Mutual Information Class Amino Acid Mutual Information $R2(c_i, Y) = \frac{1}{H(c_i, Y)}\big(H(c_i) + H(Y) - H(c_i, Y)\big)$.	Internal Mutual Information Normalized Sum of Mutual Information Redundancy $R2(c_i, c_j) = \frac{1}{H(c_i, c_j)}\big(H(c_i) + H(c_j) - H(c_i, c_j)\big)$, $SR2(c_i) = \frac{1}{n}\sum_{j=1}^{n} R2(c_i, c_j)$.

AP Cluster Clusters' Indicators: k =number of clusters, $s(K_i)$ =average co-occurrence score in cluster i, $s(K_i, K_j)$ =average co-occurrence score between cluster i and j

Average Score

$$\frac{\sum_{i=1}^{k} s(K_i)}{k}$$

Intra / Inter

$$\frac{k + \sum_{i=1}^{k} s(K_i)}{k + \sum_{x=1}^{k} \sum_{y=x+1}^{k} s(K_x, K_y)}$$

Dunn index [58]

$$\frac{2 - \max_{1 \leq x,y \leq k : x \neq y} s(K_x, K_y)}{2 - \min_{1 \leq i \leq k} s(K_i)}$$

Max Intra / Related Inter

$$\frac{s(K_x)}{\sum_{y=1}^{k} s(K_x, K_y)} where\, x\, is \max \forall s(K_i)$$

B.2 List of Definitions

A motif is a sequential substring that occurs repeatedly as a pattern throughout a set of sequences.

A sequence alignment (alignment) is the arrangement of two or more sequences by adding gaps so that the characters line up optimally. An alignment could be either global or local: global means that it is optimized from end to end; local means that it can be optimized for a localized region.

A binding site is a region in the protein that binds a specific ligand, another molecule or ion, through a chemical bond. A hypergraph is generalized graph where each edge can connect any number of vertices, this edge is called a hyperedge. Homology describes the shared ancestry; thus, sequence homology describes the sequence ancestry due to speciation event (orthlog) or duplication event (paralog). For example, homologous sequences are orthologous if they descend from the same ancestral sequence by a speciation event, i.e. when a species diverges into two different species. For example, homologous sequences are paralogous due to a duplication event, i.e. when a gene in an organism is duplicated to occupy two different places.

References

[1] F. A. Akinniyi, Andrew K. C. Wong, and D. Stacey. A new algorithm for graph monomorphism based on the projections of the product graph. *IEEE Trans on Systems, Man and Cybernetics*, pages 740–751, 1986.

[2] Bruce Alberts, Alexander Johnson, Julian Lewis, Martin Raff, Keith Roberts, and Peter Walter. *Molecular Biology of the Cell*. Garland, 4 edition, 2002. 165

[3] N. I. Aleksandrushkina and L. A. Egorova. Nucleotide makeup of the dna of thermophilic bacteria of the genus thermus. *Mikrobiologiia*, 47(2):250–252, 1978. 7

[4] J.W.A. Allen, O. Daltrop, J.M. Stevens, S.J. Ferguson, J.W.A. Allen, O. Daltrop, J.M. Stevens, and S.J. Ferguson. C-type cytochromes: diverse structures and biogenesis systems pose evolutionary problems. *Philosophical Transactions of the Royal Society of London. Series B: Biological Sciences*, 358(1429):255–266, 2003.

[5] D Altschuh, AM Lesk, AC Bloomer, and A Klug. Correlation of co-ordinated amino acid substitutions with function in viruses related to tobacco mosaic virus. *Journal of molecular biology*, 193(4):693–707, 1987.

[6] S. F. Altschul, W. Gish, W. Miller, and E. W. Myers. Basic local alignment search tool. *J. Mol. Biol.*, 215:403–410, 1990.

[7] S. F. Altschul, T. L. Madden, and A. A. Schaffer. Gapped blast and psi-blast: A new generation of protein database search programs. *Nucleic Acids Research*, 25(17):3389–3402, 1997.

[8] and Michael Kaufmann Amarendran R Subramanian and Burkhard Morgenstern. Dialign-tx: greedy and progressive approaches for segment-based multiple sequence alignment. *Algorithms Mol Biol*, 3:6, 2008. 6

[9] Antonina Andreeva, Dave Howorth, John-Marc Chandonia, Steven E Brenner, Tim JP Hubbard, Cyrus Chothia, and Alexey G Murzin. Data growth and its impact on the scop database: new developments. *Nucleic acids research*, 36(suppl 1):D419–D425, 2008. 20

[10] Mohamed Arredouani, Zhiping Yang, YaoYu Ning, Guozhong Qin, Raija Soininen, Karl Tryggvason, and Lester Kobzik. The scavenger receptor marco is required for lung defense against pneumococcal pneumonia and inhaled particles. *The Journal of experimental medicine*, 200(2):267–272, 2004.

[11] William R Atchley, Kurt R Wollenberg, Walter M Fitch, Werner Terhalle, and Andreas W Dress. Correlations among amino acid sites in bhlh protein domains: an information theoretic analysis. *Molecular biology and evolution*, 17(1):164–178, 2000.

[12] T. K. Attwood. The quest to deduce protein function from sequence: the role of pat-

tern databases. *The International Journal of Biochemistry & Cell Biology*, 32:139–155, 2000.

[13] Terri K. Attwood, Paul Bradley, Darren R. Flower, Anna Gaulton, Neil Maudling, AL Mitchell, G Moulton, A Nordle, K Paine, P Taylor, et al. Prints and its automatic supplement, preprints. *Nucleic acids research*, 31(1):400–402, 2003.

[14] Timothy L. Bailey, Mikael Boden, Fabian A. Buske, Martin Frith, Charles E. Grant, Luca Clementi, Jingyuan Ren, Wilfred W. Li, and William S. Noble. Meme suite: tools for motif discovery and searching. *Nucleic Acids Research*, 37:W202–W208, 2009. 23, 41, 65

[15] Timothy L Bailey and Charles Elkan. Unsupervised learning of multiple motifs in biopolymers using expectation maximization. *Machine learning*, 21(1-2):51–80, 1995.

[16] A Bairoch. Prosite: a dictionary of sites and patterns in proteins. *Nucleic Acids Research*, 19:2241–2245, 1991. 19, 56, 88

[17] Amos Bairoch, Rolf Apweiler, Cathy H Wu, Winona C Barker, Brigitte Boeckmann, Serenella Ferro, Elisabeth Gasteiger, Hongzhan Huang, Rodrigo Lopez, Michele Magrane, et al. The universal protein resource (uniprot). *Nucleic acids research*, 33(suppl 1):D154–D159, 2005. 19, 135

[18] S. Balla, J. Davila, and S. Rajasekaran. On the challenging instances of the planted motif problem. Technical report, 2007.

[19] Paul D Barker and Stuart J Ferguson. Still a puzzle: why is haem covalently attached in c-type cytochromes? *Structure*, 7(12):R281–R290, 1999. 154, 155

[20] Helen M Berman, John Westbrook, Zukang Feng, Gary Gilliland, TN Bhat, Helge Weissig, Ilya N Shindyalov, and Philip E Bourne. The protein data bank. *Nucleic acids research*, 28(1):235–242, 2000. 19, 144

[21] Helen M Berman, John Westbrook, Zukang Feng, Gary Gilliland, TN Bhat, Helge Weissig, Ilya N Shindyalov, and Philip E Bourne. The protein data bank. *Nucleic acids research*, 28(1):235–242, 2000.

[22] Frances C Bernstein, Thomas F Koetzle, Graheme JB Williams, Edgar F Meyer Jr, Michael D Brice, John R Rodgers, Olga Kennard, Takehiko Shimanouchi, and Mitsuo Tasumi. The protein data bank: a computer-based archival file for macromolecular structures. *Journal of molecular biology*, 112(3):535–542, 1977.

[23] Christopher M Bishop et al. *Pattern recognition and machine learning*, volume 1. springer New York, 2006. 141

[24] Karen M Black and Carmichael JA Wallace. Probing the role of the conserved β-ii turn pro-76/gly-77 of mitochondrial cytochrome c. *Biochemistry and cell biology*, 85(3):366–374, 2007. 154, 155

[25] Jacek Blazewicz, Piotr Lukasiak, and Maciej Milostan. Some operations research methods for analyzing protein sequences and structures. *4OR: A Quarterly Journal of Operations Research*, 4(2):91–123, 2006.

[26] J.D. Bloom and M.J. Glassman. Inferring stabilizing mutations from protein phylogenies: application to influenza hemagglutinin. *PLoS computational biology*, 5(4):e1000349, 2009.

[27] Philip E. Bourne and Helge Weissig. *Structural Bioinformatics*. Wiley-Liss, 2003.

[28] Dawn ME Bowdish and Siamon Gordon. Conserved domains of the class a scavenger receptors: evolution and function. *Immunological reviews*, 227(1):19–31, 2008.

[29] Sarah EJ Bowman and Kara L Bren. The chemistry and biochemistry of heme c: functional bases for covalent attachment. *Natural product reports*, 25(6):1118–1130, 2008. 154, 155

[30] B. Brejová, T. Vinar, and M. Li. Pattern discovery: Methods and software. *Introduction to Bioinformatics*, pages 491–522, 2003. 6

[31] J. Buhler and M. Tompa. Finding motifs using random projections. *J Comput Biol*, 2002. 7

[32] Jeremy Buhler and Martin Tompa. Finding motifs using random projections. *Journal of computational biology*, 9(2):225–242, 2002. 23, 41, 65

[33] Jeremy Buhler and Martin Tompa. Finding motifs using random projections. *Journal of computational biology*, 9(2):225–242, 2002.

[34] Timothy J Burch and Arthur L Haas. Site-directed mutagenesis of ubiquitin. differential roles for arginine in the interaction with ubiquitin-activating enzyme. *Biochemistry*, 33(23):7300–7308, 1994. 149, 151

[35] Lukas Burger and Erik van Nimwegen. Disentangling direct from indirect co-evolution of residues in protein alignments. *PLoS computational biology*, 6(1):e1000633, 2010. 10, 26, 134

[36] Forbes Burkowski. *STRUCTURAL BIOINFORMATICS: An Algorithmic Approach*. CRC Press, 2008.

[37] Sergiy Butenko, W. Art Chaovalitwongse, and Panos M. Pardalos. *Clustering Challenges in Biological Networks*. World Scientific, illustrated edition, 2009. 6

[38] Christopher Bystroff, Vesteinn Thorsson, and David Baker. Hmmstr: a hidden markov model for local sequence-structure correlations in proteins. *Journal of molecular biology*, 301(1):173–190, 2000. 25

[39] S. C. Chan and Andrew K. C. Wong. Synthesis and recognition of sequences. *IEEE Trans on PAMI*, 13(12):1245–1255, 1991. 7, 23, 37, 44

[40] S.C. Chan, Andrew K. C. Wong, and David K. Y. Chiu. A survey of multiple sequence comparison methods. *Bull Math Biol*, 54(4):563–598, 1992.

[41] Tak-Ming Chan, Leung-Yau Lo, Ho-Yin Sze-To, Kwong-Sak Leung, Xinshu Xiao, and Man-Hon Wong. Modeling associated protein-dna pattern discovery with unified scores. *IEEE/ACM transactions on computational biology and bioinformatics/IEEE, ACM*, 2013. 10, 26

[42] Wilfredo Colon, L. Paul Wakem, Fred Sherman, and Heinrich Roder. Identification of the predominant non-native histidine ligand in unfolded cytochrome c. *Biochemistry*, 36:12535–12541, 1997. 28, 55

[43] UniProt Consortium et al. Activities at the universal protein resource (uniprot). *Nucleic Acids Research*, 42(D1):D191–D198, 2014. 143

[44] Seth Cooper, Firas Khatib, Adrien Treuille, Janos Barbero, Jeehyung Lee, Michael Beenen, Andrew Leaver-Fay, David Baker, Zoran Popovi, and Foldit players. Predicting protein structures with a multiplayer online game. *Nature*, 466:756–760, 2010.

[45] Alison L Cuff, Ian Sillitoe, Tony Lewis, Andrew B Clegg, Robert Rentzsch, Nicholas Furnham, Marialuisa Pellegrini-Calace, David Jones, Janet Thornton, and Christine A Orengo. Extending cath: increasing coverage of the protein structure universe and linking structure with function. *Nucleic acids research*, 39(suppl 1):D420–D426, 2011. 20

[46] M.K. Das and H.K. Dai. A survey of dna motif finding algorithms. *BMC bioinformatics*, 8(Suppl 7):S21, 2007.

[47] Modan Das and Ho-Kwok Dai. A survey of dna motif finding algorithms. *BMC bioinformatics*, 8(Suppl 7):S21, 2007.

[48] Robert C Davenport, Paul A Bash, Barbara A Seaton, Martin Karplus, Gregory A Petsko, and Dagmar Ringe. Structure of the triosephosphate isomerase-phosphoglycolohydroxamate complex: an analog of the intermediate on the reaction pathway. *Biochemistry*, 30(24):5821–5826, 1991.

[49] David L Davies and Donald W Bouldin. A cluster separation measure. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (2):224–227, 1979.

[50] M. O. Dayhoff, R. M. Schwartz, and B. C. Orcutt. A model of evolutionary change in proteins. *Atlas of Protein Sequence and Structure*, 5 (3):345352, 1978. 43

[51] David de Juan, Florencio Pazos, and Alfonso Valencia. Emerging methods in protein co-evolution. *Nature Reviews Genetics*, 2013. 16, 96

[52] Elizabeth A. Dethoff, Jeetender Chugh, Anthony M. Mustoe, and Hashim M. Al-Hashimi. Functional complexity and regulation through rna dynamics. *Nature*, 482:322–330, 2012.

[53] Ivan Dikic, Soichi Wakatsuki, and Kylie J Walters. Ubiquitin-binding domainsfrom structures to functions. *Nature reviews Molecular cell biology*, 10(10):659–671, 2009. 149, 151

[54] Hieu Dinh, Sanguthevar Rajasekaran, and Jaime Davila. qpms7: A fast algorithm for finding (l, d)-motifs in dna and protein sequences. *PloS one*, 7(7):e41425, 2012. 23, 41

[55] Chuong B Do, Mahathi SP Mahabhashyam, Michael Brudno, and Serafim Batzoglou. Probcons: Probabilistic consistency-based multiple sequence alignment. *Genome research*, 15(2):330–340, 2005. 21

[56] T Doi, K-I Higashino, Y Kurihara, Y Wada, T Miyazaki, H Nakamura, S Uesugi, T Imanishi, Y Kawabe, H Itakura, et al. Charged collagen structure mediates the recognition of negatively charged macromolecules by macrophage scavenger receptors. *Journal of Biological Chemistry*, 268(3):2126–2133, 1993.

[57] Michael G Dorrington, Aoife M Roche, Sarah E Chauvin, Zhongyuan Tu, Karen L Mossman, Jeffrey N Weiser, and Dawn ME Bowdish. Marco is required for tlr2-and

nod2-mediated responses to streptococcus pneumoniae and clearance of pneumococcal colonization in the murine nasopharynx. *The Journal of Immunology*, 190(1):250–258, 2013.

[58] Joseph C Dunn. A fuzzy relative of the isodata process and its use in detecting compact well-separated clusters. 1973. 175

[59] Richard Durbin. *Biological sequence analysis: probabilistic models of proteins and nucleic acids*. Cambridge university press, 1998.

[60] Richard Durbin, Sean R. Eddy, Anders Krogh, and Graeme Mitchison. *Biological Sequence Analysis: Probabilistic models of proteins and nucleic acids*. Cambridge University Press, 1998. 6

[61] Sean R. Eddy. Profile hidden markov models. *Bioinformatics*, 14(9):755–763, 1998.

[62] K. Ellrott, J. T. Guo, V. Olman, and Y. Xu. Improvement in protein sequence-structure alignment using insertion/deletion frequency arrays. *Computer Systems Bioinformatics Conference*, 6:335–42, 2007.

[63] Robert D Finn, Jaina Mistry, John Tate, Penny Coggill, Andreas Heger, Joanne E Pollington, O Luke Gavin, Prasad Gunasekaran, Goran Ceric, Kristoffer Forslund, et al. The pfam protein families database. *Nucleic acids research*, 38(suppl 1):D211–D222, 2010. 19, 20, 135, 143

[64] W.R. Fisher, H. Taniuchi, and C.B. Anfinsen. On the role of heme in the formation of the structure of cytochrome c. *Journal of Biological Chemistry*, 248(9):3188–3195, 1973.

[65] Mark S. Forman, John Q. Trojanowski, and Virginia M-Y Lee. Neurodegenerative diseases: a decade of discoveries paves the way for therapeutic breakthroughs. *Nature medicine*, 10.10:1055–1063, 2004.

[66] M. Frances and A. Litman. On covering problems of codes. *Theory of Computing Systems*, 30:113–119, 1997.

[67] Alexandre P Francisco, Sophie Schbath, Ana T Freitas, and Arlindo L Oliveira. Using graph modularity analysis to identify transcription factor binding sites. In *Bioinformatics and Biomedicine Workshops (BIBMW), 2010 IEEE International Conference on*, pages 19–26. IEEE, 2010.

[68] Zoey L Fredericks and Gary J Pielak. Exploring the interface between the n-and c-terminal helixes of cytochrome c by random mutagenesis within the c-terminal helix. *Biochemistry*, 32(3):929–936, 1993. 154, 155

[69] Martin C Frith, Ulla Hansen, John L Spouge, and Zhiping Weng. Finding functional sequence elements by multiple local alignment. *Nucleic acids research*, 32(1):189–200, 2004. 7

[70] Shuba Gopal, Anne Haake, Rhys Price Jones, and Paul Tymann. *Bioinformatics: A computing perspective*. McGraw-Hill Science/Engineering/Math, 2008.

[71] Francesc X Guix, Gerard Ill-Raga, Ramona Bravo, Tadashi Nakaya, Gianni de Fabritiis, Mireia Coma, Gian Pietro Miscione, Jordi Villà-Freixa, Toshiharu Suzuki, Xavier Fernàndez-Busquets, et al. Amyloid-dependent triosephosphate isomerase

nitrotyrosination induces glycation and tau fibrillation. *Brain*, 132(5):1335–1345, 2009.

[72] Francesc X Guix, Gerard Ill-Raga, Ramona Bravo, Tadashi Nakaya, Gianni de Fabritiis, Mireia Coma, Gian Pietro Miscione, Jordi Villà-Freixa, Toshiharu Suzuki, Xavier Fernàndez-Busquets, et al. Amyloid-dependent triosephosphate isomerase nitrotyrosination induces glycation and tau fibrillation. *Brain*, 132(5):1335–1345, 2009.

[73] Isabelle Guyon and André Elisseeff. An introduction to variable and feature selection. *The Journal of Machine Learning Research*, 3:1157–1182, 2003. 25

[74] Stephen J Hagen, Ramil F Latypov, Dimitry A Dolgikh, and Heinrich Roder. Rapid intrachain binding of histidine-26 and histidine-33 to heme in unfolded ferrocytochrome c. *Biochemistry*, 41(4):1372–1380, 2002. 154, 155

[75] Patrice P. Hamel, Beth Welty Dreyfuss, Zhiyi Xie, Stphane T. Gabilly, and Sabeeha Merchant. Essential histidine and tryptophan residues in ccsa, a system ii polytopic cytochrome c biogenesis protein. *The Journal of Biological Chemistry*, 278:2593–2603, 2003. 57

[76] Thomas K Harris, Chitrananda Abeygunawardana, and Albert S Mildvan. Nmr studies of the role of hydrogen bonding in the mechanism of triosephosphate isomerase. *Biochemistry*, 36(48):14661–14675, 1997.

[77] J.G. Henikoff and S. Henikoff. Blocks database and its applications. *Methods in Enzymology*, 266:88–105, 1996. 22, 41, 65

[78] S. Henikoff, J. G. Henikoff, and S. Pietrokovski. Blocks+: a non-redundant database of protein alignment blocks derived from multiple compilations. *Bioinformatics*, 15(6):471–479, 1995.

[79] S Henikoff and JG Henikoff. Amino acid substitution matrices from protein blocks. *Proceedings of the National Academy of Sciences of the United States of America*, 89 (22):109159, 1992. 43

[80] Gerald Z Hertz and Gary D. Stormo. Identifying dna and protein patterns with statistically significant alignments of multiple sequences. *Bioinformatics*, 15(7):563–577, 1999. 41, 65

[81] Yoichi Takenaka Hideya Kawaji and Hideo Matsuda. Graph-based clustering for finding distant relationships in a large set of protein sequences. *Bioinformatics*, 20:243–252, 2004. 7

[82] Daniela Hoeller and Ivan Dikic. Targeting the ubiquitin system in cancer therapy. *Nature*, 458(7237):438–444, 2009. 6

[83] Kuo-Ying Huang. Ubiquitin conformational dynamics and hydration shell dynamics by solid state nmr. 2011.

[84] JP Hughes, S Rees, SB Kalindjian, and KL Philpott. Principles of early drug discovery. *Br J Pharmacol*, 162(6):1239–1249, 2011.

[85] Fumiyo Ikeda and Ivan Dikic. Atypical ubiquitin chains: new molecular signals. *EMBO reports*, 9(6):536–542, 2008. 61

[86] Anil K Jain. Data clustering: 50 years beyond k-means. *Pattern Recognition Letters*, 31(8):651–666, 2010. 143

[87] Jong Cheol Jeong, Xiaotong Lin, and Xue-Wen Chen. On position-specific scoring matrix for protein function prediction. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 8(2):308–315, 2011. 7

[88] DLANE Joseph, Gregory A Petsko, and Martin Karplus. Anatomy of a conformational change: hinged" lid" motion of the triosephosphate isomerase loop. *Science*, 249(4975):1425–1428, 1990. 63

[89] Tracy M Josephs, Matthew D Liptak, Gillian Hughes, Alexandra Lo, Rebecca M Smith, Sigurd M Wilbanks, Kara L Bren, and Elizabeth C Ledgerwood. Conformational change and human cytochrome c function: mutation of residue 41 modulates caspase activation and destabilizes met-80 coordination. *JBIC Journal of Biological Inorganic Chemistry*, 18(3):289–297, 2013. 154, 155

[90] Kevin Karplus, Christian Barrett, and Richard Hughey. Hidden markov models for detecting remote protein homologies. *Bioinformatics*, 14(10):846–856, 1998.

[91] Itamar Kass and Amnon Horovitz. Mapping pathways of allosteric communication in groel by analysis of correlated mutations. *Proteins: Structure, Function, and Bioinformatics*, 48(4):611–617, 2002.

[92] Leonard Kaufman and Peter Rousseeuw. *Clustering by means of medoids*. North-Holland, 1987.

[93] Edward Casey Kenley, Lyles Kirk, and Young-Rae Cho. Differentiating party and date hubs in protein interaction networks using semantic similarity measures. In *Proceedings of the 2nd ACM Conference on Bioinformatics, Computational Biology and Biomedicine*, pages 641–645. ACM, 2011. 10, 26

[94] HT Kim, Kim, Lledias, Scaglione Kisselev, Skowyra, Gygi, and Goldberg. Certain pairs of ubiquitin-conjugating enzymes (e2s) and ubiquitin-protein ligases (e3s) synthesize condegradable forked ubiquitin chains containing all possible isopeptide linkages. *The Journal of biological chemistry*, 282 (24):17375–86, 2007. 61

[95] Akira R Kinjo and Haruki Nakamura. Composite structural motifs of binding sites for delineating biological functions of proteins. *PloS one*, 7(2):e31437, 2012.

[96] T Kirisako, K Kamei, Murata; Kato, Fukumoto, Kanie, Sano, and Tokunaga. A ubiquitin ligase complex assembles linear polyubiquitin chains. *The EMBO journal*, 25 (20):4877–87, 2006.

[97] Oleksandr Kokhan, Colin A Wraight, and Emad Tajkhorshid. The binding interface of cytochrome c and cytochrome c1 in the bc1 complex: Rationalizing the role of key residues. *Biophysical journal*, 99(8):2647–2656, 2010. 153

[98] Daphne Koller and Mehran Sahami. Toward optimal feature selection. 1996. 25

[99] Georg Kraal, Luc JW van der Laan, Outi Elomaa, and Karl Tryggvason. The macrophage receptor marco. *Microbes and infection*, 2(3):313–316, 2000.

[100] Robert G Kranz, Cynthia Richard-Fogal, John-Stephen Taylor, and Elaine R Frawley. Cytochrome c biogenesis: mechanisms for covalent modifications and trafficking

of heme and for heme-iron redox control. *Microbiology and molecular biology reviews*, 73(3):510–528, 2009. 55

[101] Robert G. Kranz, Cynthia Richard-Fogal, John-Stephen Taylor, and Elaine R. Frawley. Cytochrome c biogenesis: Mechanisms for covalent modifications and trafficking of heme and for heme-iron redox control. *Microbiology and Moelcular Biology Review*, 73(3):510–528, 2009.

[102] Monty Krieger. Molecular flypaper and atherosclerosis: structure of the macrophage scavenger receptor. *Trends in biochemical sciences*, 17(4):141–146, 1992.

[103] R. Kuang, IE EUGENE, K. Wang, K. Wang, M. Siddiqi, Y. Freund, and C. Leslie. Profile-based string kernels for remote homology detection and motif extraction. *Journal of bioinformatics and computational biology*, 3(03):527–550, 2005.

[104] P.P. Kuksa. 2d similarity kernels for biological sequence classification. pages 15–20, 2012.

[105] P.P. Kuksa and V. Pavlovic. Efficient evaluation of large sequence kernels. pages 759–767, 2012.

[106] Inari Kursula, Sanna Partanen, Anne-Marie Lambeir, Dmitry M Antonov, Koen Augustyns, and Rik K Wierenga. Structural determinants for ligand binding and catalysis of triosephosphate isomerase. *European Journal of Biochemistry*, 268(19):5189–5196, 2001. 62

[107] Inari Kursula and Rik K Wierenga. Crystal structure of triosephosphate isomerase

complexed with 2-phosphoglycolate at 0.83-å resolution. *Journal of Biological Chemistry*, 278(11):9544–9551, 2003. 63

[108] MA Larkin, Gordon Blackshields, NP Brown, R Chenna, Paul A McGettigan, Hamish McWilliam, Franck Valentin, Iain M Wallace, Andreas Wilm, Rodrigo Lopez, et al. Clustal w and clustal x version 2.0. *Bioinformatics*, 23(21):2947–2948, 2007. 20

[109] Charles E Lawrence, Stephen F Altschul, Mark S Boguski, Jun S Liu, Andrew F Neuwald, and John C Wootton. Detecting subtle sequence signals: a gibbs sampling strategy for multiple alignment. *SCIENCE-NEW YORK THEN WASHINGTON-*, 262:208–208, 1993. 23, 41, 65

[110] E.-S.A. Lee, S. Fung, Ho-Yin Sze-To, and A.K.C. Wong. Confirming biological significance of co-occurrence clusters of aligned pattern clusters. In *Bioinformatics and Biomedicine (BIBM), 2013 IEEE International Conference on*, pages 422–427, Dec 2013. 143

[111] En-Shiun Annie Lee, Sanderz Fung, Ho-Yin Sze-To, and Andrew K. C. Wong. Confirming biological significance of co-occurrence clusters of aligned pattern clusters. *BIBM*, page (To Appear), 2013. 68

[112] En-Shiun Annie Lee and Andrew K. C. Wong. Synthesizing aligned random pattern digraphs from protein sequence patterns. *Bioinformatics and Biomedicine Workshops (BIBMW)*, pages pp. 178 – 185, 2011.

[113] En-Shiun Annie Lee and Andrew K. C. Wong. Identifying protein binding functionality of protein families by aligned pattern clusters. *IEEE International Conference on Bioinformatics and Biomedicine*, 2012. 110, 111

[114] En-Shiun Annie Lee and Andrew K. C. Wong. Classifying proteins by amino acid variations of sequential patterns. *BCB 13*, page To Appear, September 22 - 25, 2013, 2013. Washington, DC, USA. 68

[115] En-Shiun Annie Lee and Andrew K. C. Wong. Revealing binding segments in protein families using aligned pattern clusters. *Proteome Science*, 2013. 8, 11, 97, 98, 135, 138

[116] En-Shiun Annie Lee and Andrew KC Wong. Ranking and compacting binding segments of protein families using aligned pattern clusters. *Proteome Science*, 11(Suppl 1):S8, 2013. 10, 34

[117] C. Leslie and R. Kuang. Fast string kernels using inexact matching for protein sequences. *The Journal of Machine Learning Research*, 5:1435–1455, 2004.

[118] Christina S Leslie, Eleazar Eskin, Adiel Cohen, Jason Weston, and William Stafford Noble. Mismatch string kernels for discriminative protein classification. *Bioinformatics*, 20(4):467–476, 2004.

[119] Kwong-Sak Leung, Ka-Chun Wong, Tak-Ming Chan, Man-Hon Wong, Kin-Hong Lee, Chi-Kong Lau, and Stephen KW Tsui. Discovering protein–dna binding sequence patterns using association rule mining. *Nucleic acids research*, 38(19):6324–6337, 2010. 10, 26

[120] M. Li, B. Ma, and L. Wang. Finding similar regions in many strings. *Journal of Computer and System Sciences*, 65:73–96, 2002. 7

[121] Ming Li, Bin Ma, and Lusheng Wang. Finding similar regions in many strings. In *Proceedings of the thirty-first annual ACM symposium on Theory of computing*, pages 473–482. ACM, 1999. 22

[122] Weizhong Li, Lukasz Jaroszewski, and Adam Godzik. Clustering of highly homologous sequences to reduce the size of large protein databases. *Bioinformatics*, 17(3):282–283, 2001.

[123] Xiao-Li Li, Soon-Heng Tan, Chuan-Sheng Foo, See-Kiong Ng, et al. Interaction graph mining for protein complexes using local clique merging. *GENOME INFORMATICS SERIES*, 16(2):260, 2005. 10, 26

[124] Olivier Lichtarge, Henry R Bourne, and Fred E Cohen. An evolutionary trace method defines binding surfaces common to protein families. *Journal of molecular biology*, 257(2):342–358, 1996. 10, 18, 26, 133, 134

[125] Steve W Lockless and Rama Ranganathan. Evolutionarily conserved pathways of energetic connectivity in protein families. *Science*, 286(5438):295–299, 1999.

[126] Harvey Lodish, Arnold Berk, S. Lawrence Zipursky, Paul Matsudaira, David Baltimore, and James E. Darnell. *Molecular Cell Biology*. Sinauer Associates Inc., Sunderland, MA, 2000.

[127] Gordon V Louie, Gary J Pielak, Michael Smith, and Gary D Brayer. Role of phenylalanine-82 in yeast iso-1-cytochrome c and remote conformational changes

induced by a serine residue at this position. *Biochemistry*, 27(20):7870–7876, 1988. 154, 155

[128] Jeffrey M Macdonald, Arthur L Haas, and Robert E London. Novel mechanism of surface catalysis of protein adduct formation nmr studies of the acetylation of ubiquitin. *Journal of Biological Chemistry*, 275(41):31908–31913, 2000.

[129] Srinivasan Madabushi, Alecia K Gross, Anne Philippi, Elaine C Meng, Theodore G Wensel, and Olivier Lichtarge. Evolutionary trace of g protein-coupled receptors reveals clusters of residues that determine global and class-specific functions. *Journal of Biological Chemistry*, 279(9):8126–8132, 2004. 10, 26, 134

[130] M Merced Malabanan, Lucia Nitsch-Velasquez, Tina L Amyes, and John P Richard. Magnitude and origin of the enhanced basicity of the catalytic glutamate of triosephosphate isomerase. *Journal of the American Chemical Society*, 135(16):5978–5981, 2013. 63

[131] I. Mandoiu and A. Zelikovsky. *Bioinformatics Algorithms: Techniques and Applications*. Wiley Series in Bioinformatics. Wiley, 2008. 7

[132] Jean-Claude Martinou, Solange Desagher, and Bruno Antonsson. Cytochrome c release from mitochondria: all or nothing. *Nature Cell Biology*, 2:E41–E43, 2000.

[133] Edward M. McCreight. A space-economical suffix tree construction algorithm. *Journal of the ACM*, 23(2):262–272, 1976.

[134] Sanzo Miyazawa. Prediction of contact residue pairs based on co-substitution between sites in protein structures. *PloS one*, 8(1):e54252, 2013.

[135] Faruck Morcos, Andrea Pagnani, Bryan Lunt, Arianna Bertolino, Debora S Marks, Chris Sander, Riccardo Zecchina, José N Onuchic, Terence Hwa, and Martin Weigt. Direct-coupling analysis of residue coevolution captures native contacts across many protein families. *Proceedings of the National Academy of Sciences*, 108(49):E1293–E1301, 2011. 10, 26, 134

[136] Saul B. Needleman and Christian D. Wunsch. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology*, 48 (3):443–53, 1970. 41

[137] Pauline C Ng and Steven Henikoff. Predicting the effects of amino acid substitutions on protein function. *Annu. Rev. Genomics Hum. Genet.*, 7:61–80, 2006. 96

[138] W.S. Noble. Mismatch string kernels for svm protein classification. 2008.

[139] Cédric Notredame, Desmond G Higgins, and Jaap Heringa. T-coffee: A novel method for fast and accurate multiple sequence alignment. *Journal of molecular biology*, 302(1):205–217, 2000. 6, 20, 134

[140] Cédric Notredame, Desmond G Higgins, and Jaap Heringa. T-coffee: A novel method for fast and accurate multiple sequence alignment. *Journal of molecular biology*, 302(1):205–217, 2000.

[141] Alberto Paccanaro, James A Casbon, and Mansoor AS Saqi. Spectral clustering of protein sequences. *Nucleic acids research*, 34(5):1571–1580, 2006. 8

[142] Rupali Patwardhan, Haixu Tang, Sun Kim, and Mehmet Dalkilic. An approximate

de bruijn graph approach to multiple local alignment and motif discovery in protein sequences. *Data Mining and Bioinformatics*, 4316:158–169, 2006. 7

[143] Giulio Pavesi, Paolo Mereghetti, Giancarlo Mauri, and Graziano Pesole. Weeder web: discovery of transcription factor binding sites in a set of sequences from co-regulated genes. *Nucleic acids research*, 32(suppl 2):W199–W203, 2004. 23

[144] Junmin Peng, Daniel Schwartz, Joshua E Elias, Carson C Thoreen, Dongmei Cheng, Gerald Marsischky, Jeroen Roelofs, Daniel Finley, and Steven P Gygi. A proteomics approach to understanding protein ubiquitination. *Nature biotechnology*, 21(8):921–926, 2003.

[145] Julian Peto. Cancer epidemiology in the last century and the next decade. *Nature*, 411.6835:390–395, 2001.

[146] Gregory A Petsko and Dagmar Ringe. *Protein structure and function*. New Science Press, 2004.

[147] Eric F Pettersen, Thomas D Goddard, Conrad C Huang, Gregory S Couch, Daniel M Greenblatt, Elaine C Meng, and Thomas E Ferrin. Ucsf chimera-a visualization system for exploratory research and analysis. *Journal of computational chemistry*, 25(13):1605–1612, 2004. 144

[148] P. Pevzner and S. Sze. Combinatorial approaches to finding subtle signals in dna strings. *In Proc. ISMB*, 2000:269–278, 2000. 7

[149] Pavel A Pevzner, Sing-Hoi Sze, et al. Combinatorial approaches to finding subtle signals in dna sequences. In *ISMB*, pages 269–278, 2000. 23

[150] Peter Piot, Michael Bartos, Peter D. Ghys, Neff Walker, and Bernhard Schwartlnder. The global impact of hiv/aids. *Nature*, 410.6831:968–973, 2001.

[151] Nadia Pisanti, Maxime Crochemore, Roberto Grossi, and Marie-France Sagot. Bases of motifs for generating repeated patterns with wild cards. *IEEE/ACM Transactions on Computational BIology and Bioinformatics*, 2(1):40–50, 2005.

[152] Nick Platt, Siamon Gordon, et al. Is the class a macrophage scavenger receptor (sr-a) multifunctional? the mouse's tale. *Journal of Clinical Investigation*, 108(5):649–654, 2001.

[153] C.P. Ponting. Issues in predicting protein function from sequence. *Briefings in bioinformatics*, 2(1):19–29, 2001.

[154] N. Provart. Motif and profile analysis. Bio472 lecture of, University of Toronto, 14 Mar 2007 2007.

[155] T. A. Reichert, D. N. Cohen, and Andrew K. C. Wong. An application of information theory to genetic mutations and the matching of polypeptide sequences. *Journal of Theoretical Biology*, 42:245–261, 1973.

[156] David N Reshef, Yakir A Reshef, Hilary K Finucane, Sharon R Grossman, Gilean McVean, Peter J Turnbaugh, Eric S Lander, Michael Mitzenmacher, and Pardis C Sabeti. Detecting novel associations in large data sets. *science*, 334(6062):1518–1524, 2011. 25

[157] Claudia Rodríguez-Almazán, Rodrigo Arreola, David Rodríguez-Larrea, Beatriz Aguirre-López, Marietta Tuena de Gómez-Puyou, Ruy Pérez-Montfort, Miguel

Costas, Armando Gómez-Puyou, and Alfredo Torres-Larios. Structural basis of human triosephosphate isomerase deficiency mutation e104d is related to alterations of a conserved water network at the dimer interface. *Journal of Biological Chemistry*, 283(34):23254–23263, 2008. 62

[158] Federico I Rosell, Thomas R Harris, Dean P Hildebrand, Susanne Döpner, Peter Hildebrandt, and A Grant Mauk. Characterization of an alkaline transition intermediate stabilized in the phe82trp variant of yeast iso-1-cytochrome c. *Biochemistry*, 39(30):9047–9054, 2000.

[159] Steward Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall, 1995.

[160] H. Saigo, J.P. Vert, N. Ueda, and T. Akutsu. Protein homology detection using string alignment kernels. *Bioinformatics*, 20(11):1682–1689, 2004.

[161] VB Sampson, T Alleyne, and D Ashe. Probing the specifics of substrate binding for cytochrome c oxidase a computer assisted approach. *West Indian Medical Journal*, 58(1), 2009. 153

[162] Geir Kjetil Sandve and Finn Drablos. A survey of motif discovery methods in an integrated framework. *Biology Direct*, 1:11, 2006.

[163] R Sanishvili, KW Volz, EM Westbrook, and E Margoliash. The low ionic strength crystal structure of horse cytochrome c at 2.1 å resolution and comparison with its high ionic strength counterpart. *Structure*, 3(7):707–716, 1995. 154, 155

[164] J. Michael Sauder, Jonathan W. Arthur, and Roland L. Dunbrack Jr. Large-scale comparison of protein sequence alignment algorithms with structure alignment. *Proteins: Structure, Function, and Bioinformatics*, 40(1):6–22, 2000.

[165] Lawrence K Saul, Kilian Q Weinberger, Jihun H Ham, Fei Sha, and Daniel D Lee. Spectral methods for dimensionality reduction. *Semisupervised learning*, pages 293–308, 2006. 25

[166] A Schejter, TI Koshy, TL Luntz, R Sanishvili, I Vig, and E Margoliash. Effects of mutating asn-52 to isoleucine on the haem-linked properties of cytochrome c. *Biochem. J*, 302:95–101, 1994. 154, 155

[167] Tamar Schlick. *Molecular Modeling and Simulation*. Springer-Verlag New York, Inc., 2002.

[168] Marcel H. Schulz, David Weese, and Andreas Doring Tobias Rausch, Knut Reinert, and Martin Vingron. Fast and adaptive variable order markov chain construction. *Lecture Notes in Computer Science*, 5251:306–317, 2008.

[169] Claude Elwood Shannon. A mathematical theory of communication. *ACM SIGMOBILE Mobile Computing and Communications Review*, 5(1):3–55, 2001. 100, 103

[170] Hardik A. Sheth and Sun Kim. Motif discovery for proteins using subsequence clustering. *BioKDD*, 2005.

[171] I. N. Shindyalov and P. E. Bourne. Protein structure alignment by incremental combinatorial extension (ce) of the optimal path. *Protein Engineering*, 9:739–747, 1998.

[172] Christian JA Sigrist, Lorenzo Cerutti, Edouard De Castro, Petra S Langendijk-Genevaux, Virginie Bulliard, Amos Bairoch, and Nicolas Hulo. Prosite, a protein domain database for functional characterization and annotation. *Nucleic acids research*, 38(suppl 1):D161–D166, 2010. 56, 88

[173] Saurabh Sinha and Martin Tompa. Ymf: a program for discovery of novel transcription factor binding sites by statistical overrepresentation. *Nucleic acids research*, 31(13):3586–3588, 2003. 23

[174] Temple F. Smith and Michael S. Waterman. Dentification of common molecular subsequences. *Journal of Molecular Biology*, 147:195–197, 1981. 41

[175] Erik L.L. Sonnhammer, Sean R. Eddy, and Richard Durbin. Pfam: A comprehensive database of protein domain families based on seed alignments. *PROTEINS: Structure, Function, and Genetics*, 28:405–420, 1997. 7, 23, 56, 88

[176] Terry Speed. Motifs, profiles and hidden markov models. Presentation, Melbourne Bioinformatics Course, September 2003.

[177] A. Statnikov, C.F. Aliferis, and D.P. Hardin. *A Gentle Introduction to Support Vector Machines in Biomedicine: Theory and Methods*, volume 1. World Scientific Publishing Company Incorporated, 2011.

[178] J.M. Stevens, O. Daltrop, J.W.A. Allen, and S.J. Ferguson. C-type cytochrome formation: chemical and biological enigmas. *Accounts of chemical research*, 37(12):999–1007, 2004. 55

[179] Gürol M Süel, Steve W Lockless, Mark A Wall, and Rama Ranganathan. Evolutionarily conserved networks of residues mediate allosteric communication in proteins. *Nature Structural & Molecular Biology*, 10(1):59–69, 2002.

[180] and Zhiping Wang Sun Kim and Mehmet Dalkilic. igibbs: Improving gibbs motif sampler for proteins by sequence clustering and iterative pattern sampling. *PROTEINS: Structure, Function, and Bioinformatics*, 66:67811–6, 2007.

[181] Hiroshi Suzuki, Yukiko Kurihara, Motohiro Takeya, Nobuo Kamada, Motoyukl Kataoka, Kouichi Jishage, Otoya Ueda, Hisashl Sakaguchi, Takayuki Higashi, Tsukasa Suzuki, et al. A role for macrophage scavenger receptors in atherosclerosis and susceptibility to infection. 1997.

[182] Tsunehiro Takano and Richard E Dickerson. Redox conformation changes in refined tuna cytochrome c. *Proceedings of the National Academy of Sciences*, 77(11):6371–6375, 1980. 153

[183] Tsunehiro Takano and Richard E Dickerson. Conformation change of cytochrome c: I. ferrocytochrome c structure refined at 1· 5 å resolution. *Journal of molecular biology*, 153(1):79–94, 1981. 154, 155

[184] Pang-Ning Tan, Michael Steinbach, and Vipin Kumar. *Introduction to Data Mining*. Addison-Wesley, 2006. 139

[185] Denis Tempé, Muriel Brengues, Pauline Mayonove, Hayat Bensaad, Céline Lacrouts, and May C Morris. The alpha helix of ubiquitin interacts with yeast cyclin-dependent kinase subunit cks1. *Biochemistry*, 46(1):45–54, 2007. 151, 152

[186] Julie D Thompson, Desmond G Higgins, and Toby J Gibson. Clustal w: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic acids research*, 22(22):4673–4680, 1994. 6, 134

[187] Julie D Thompson, Benjamin Linard, Odile Lecompte, and Olivier Poch. A comprehensive benchmark study of multiple sequence alignment methods: current challenges and future perspectives. *PloS one*, 6(3):e18093, 2011. 134

[188] Julie D. Thompson, Frédéric Plewniak, and Olivier Poch. Balibase: a benchmark alignment database for the evaluation of multiple alignment programs. *Bioinformatics*, 15(1):87–88, 1999. 22

[189] M. Tompa, N. Li, T.L. Bailey, G.M. Church, B. De Moor, E. Eskin, A.V. Favorov, M.C. Frith, Y. Fu, W.J. Kent, et al. Assessing computational tools for the discovery of transcription factor binding sites. *Nature biotechnology*, 23(1):137–144, 2005.

[190] Martin Tompa, Nan Li, Timothy L Bailey, George M Church, Bart De Moor, Eleazar Eskin, Alexander V Favorov, Martin C Frith, Yutao Fu, W James Kent, et al. Assessing computational tools for the discovery of transcription factor binding sites. *Nature biotechnology*, 23(1):137–144, 2005. 23

[191] Eduardo Torres, J Victor Sandoval, Federico I Rosell, A Grant Mauk, and Rafael Vazquez-Duhalt. Site-directed mutagenesis improves the biocatalytic activity of iso-1-cytochrome c in polycyclic hydrocarbon oxidation. *Enzyme and microbial technology*, 17(11):1014–1020, 1995.

[192] E. Ukkonen. On-line construction of suffix trees. *Algorithmica*, 14(3):249–260, 1995.

[193] A. V. Ulyanov and G. D. Stormo. Multi-alphabet consensus algorithm for identification of low specificity protein-dna interactions. *Nucleic Acids Res*, 23(8):1434–1440, 1995.

[194] Rajaram Venkatesan, Markus Alahuhta, Petri M Pihko, and Rik K Wierenga. High resolution crystal structures of triosephosphate isomerase complexed with its suicide inhibitors: The conformational flexibility of the catalytic glutamate in its closed, liganded active site. *Protein Science*, 20(8):1387–1397, 2011. 62

[195] S Vijay-Kumar, CE Bugg, KD Wilkinson, RD Vierstra, PM Hatfield, and WJ Cook. Comparison of the three-dimensional structures of human, yeast, and oat ubiquitin. *Journal of Biological Chemistry*, 262(13):6396–6399, 1987. 152

[196] Senadhi Vijay-Kumar, Charles E Bugg, Keith D Wilkinson, and William J Cook. Three-dimensional structure of ubiquitin at 2.8 a resolution. *Proceedings of the National Academy of Sciences*, 82(11):3582–3585, 1985. 148, 151

[197] Ulrike Von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 17(4):395–416, 2007. 138, 140

[198] Kiri Wagstaff, Claire Cardie, Seth Rogers, and Stefan Schrödl. Constrained k-means clustering with background knowledge. In *MACHINE LEARNING-INTERNATIONAL WORKSHOP THEN CONFERENCE-*, pages 577–584, 2001.

[199] Carmichael JA Wallace and Ian Clark-Lewis. A rationale for the absolute conserva-

tion of asn70 and pro71 in mitochondrial cytochromes c suggested by protein engineering. *Biochemistry*, 36(48):14733–14740, 1997. 154, 155

[200] CJ Wallace, P Mascagni, BT Chait, JF Collawn, Y Paterson, AE Proudfoot, and SB Kent. Substitutions engineered by chemical synthesis at three conserved sites in mitochondrial cytochrome c. thermodynamic and functional consequences. *Journal of Biological Chemistry*, 264(26):15199–15209, 1989. 154, 155

[201] D Wang and A Wong. Classification of discrete data with feature space transformation. *Automatic Control, IEEE Transactions on*, 24(3):434–437, 1979. 25

[202] LuShen Wang and Tao Jiang. On the complexity of multiple sequence alignment. *Journal of Computational Biology*, 1(4):337–348, 1994. 6

[203] Lusheng Wang and Tao Jiang. On the complexity of multiple sequence alignment. *Journal of computational biology*, 1(4):337–348, 1994. 134

[204] Martin Weigt, Robert A White, Hendrik Szurmant, James A Hoch, and Terence Hwa. Identification of direct residue contacts in protein–protein interaction by message passing. *Proceedings of the National Academy of Sciences*, 106(1):67–72, 2009. 10, 26, 134

[205] Martin Weigt, Robert A White, Hendrik Szurmant, James A Hoch, and Terence Hwa. Identification of direct residue contacts in protein–protein interaction by message passing. *Proceedings of the National Academy of Sciences*, 106(1):67–72, 2009.

[206] Peter Weiner. Linear pattern matching algorithm. *14th Annual IEEE Symposium on Switching and Automata Theory*, pages 1–11, 1973.

[207] Fiona J Whelan, Conor J Meehan, G Brian Golding, Brendan J McConkey, and Dawn M E Bowdish. The evolution of the class a scavenger receptors. *BMC Evolutionary Biology*, 12:227, 2012.

[208] RK Wierenga, EG Kapetaniou, and R Venkatesan. Triosephosphate isomerase: a highly evolved biocatalyst. *Cellular and molecular life sciences*, 67(23):3961–3982, 2010. 63

[209] Kipling W Will, Brent D Mishler, and Quentin D Wheeler. The perils of dna barcoding and the need for integrative taxonomy. *Systematic Biology*, 54(5):844–851, 2005. 8

[210] Henk Wolda. Similarity indices, sample size and diversity. *Oecologia*, 50(3):296–302, 1981.

[211] Henk Wolda. Similarity indices, sample size and diversity. *Oecologia*, 50(3):296–302, 1981.

[212] Andrew K. C. Wong, Wai-Ho Au, and Keith C. C. Chan. Discovering high-order patterns of gene expression levels. *Journal of Computational Biology*, 15(6):625–637, 2008.

[213] Andrew K. C. Wong, David K. Y. Chiu, and S. C. Chan. Pattern detection in biomolecules using synthesized random sequence. *Journal of Pattern Recognition*, 29:9:1581–1586, 1995. 7, 23, 37

[214] Andrew K. C. Wong, T. S. Liu, and C. C. Wang. Statistical analysis of residue variability in cytochrome c. *Journal of Molecular Biology*, 102:287–295, 1976. 109

[215] Andrew K. C. Wong, Yang Wang, and Gary C. L. Li. Simultaneous pattern and data clustering for pattern cluster analysis. *IEEE Trans. Knowl. Data Engineering*, 20(7):911–923, 2008.

[216] Andrew K. C. Wong, Dennis Zhuang, Gary C.L. Li, and En-Shiun Annie Lee. Discovery of non-induced patterns from sequences. *Pattern Recognition in Bioinformatics*, pages 149–160, 2010.

[217] Andrew KC Wong and PK Sahoo. A gray-level threshold selection method based on maximum entropy principle. *Systems, Man and Cybernetics, IEEE Transactions on*, 19(4):866–871, 1989.

[218] Andrew KC Wong, Dennis Zhuang, Gary CL Li, and E-SA Lee. Discovery of delta closed patterns and noninduced patterns from sequences. *Knowledge and Data Engineering, IEEE Transactions on*, 24(8):1408–1421, 2012. 11, 135, 137

[219] Andrew KC Wong, Dennis Zhuang, Gary CL Li, and E-SA Lee. Discovery of delta closed patterns and noninduced patterns from sequences. *Knowledge and Data Engineering, IEEE Transactions on*, 24(8):1408–1421, 2012.

[220] Andrew K..C. Wong, Dennis Zhuang, Gary C.L. Li, and En-Shiun Annie Lee. Discovery of delta closed patterns and non-induced patterns from sequences. *IEEE Transactions on Knowledge and Data Engineering Journal*, 24(8):1408–1421, 2012. 27, 28, 31, 33, 39

[221] Xuhua Xia. Position weight matrix, gibbs sampler, and the associated significance tests in motif characterization and prediction. *Scientifica*, 2012, 2012. 135

[222] Xuhua Xia. Position weight matrix, gibbs sampler, and the associated significance tests in motif characterization and prediction. *Scientifica*, 2012, 2012.

[223] P; Peng Xu. Characterization of polyubiquitin chain structure by middle-down mass spectrometry. *Analytical chemistry*, 80(9):3438–44, 2008. 61

[224] S.R. Yeh and D.L. Rousseau. Folding intermediates in cytochrome c. *Nature Structural & Molecular Biology*, 5(3):222–228, 1998.

[225] Xiaofei Yu, Huanfa Yi, Chunqing Guo, Daming Zuo, Yanping Wang, Hyung L Kim, John R Subjeck, and Xiang-Yang Wang. Pattern recognition scavenger receptor cd204 attenuates toll-like receptor 4-induced nf-κb activation by directly inhibiting ubiquitination of tumor necrosis factor (tnf) receptor-associated factor 6. *Journal of Biological Chemistry*, 286(21):18795–18806, 2011.

[226] Sobia Zaidi, Md Imtaiyaz Hassan, Asimul Islam, and Faizan Ahmad. The role of key residues in structure, function, and stability of cytochrome-c. *Cellular and Molecular Life Sciences*, 71(2):229–255, 2014. 153, 154, 155

[227] C. Zhang and Andrew K. C. Wong. A genetic algorithm for multiple sequence alignment. *Computer Application in Biosciences*, 13(6):565–581, 1997.

[228] Yang Zhang and Jeffrey Skolnick. Scoring function for automated assessment of protein structure template quality. *Proteins: Structure, Function, and Bioinformatics*, 57 (4):702–710, 2004.

[229] J. Zhou, J. Zheng, and S. Jiang. Molecular simulation studies of the orientation and

conformation of cytochrome c adsorbed on self-assembled monolayers. *The Journal of Physical Chemistry B*, 108(45):17418–17424, 2004.