GIBBS SAMPLING FOR GAPPED MOTIF DISCOVERY IN PROTEINS

by

James Wagner

B. Sc., University of Alberta, 2005

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE
in the School
of
Computing Science

SIMON FRASER UNIVERSITY
Fall 2008

APPROVAL

Name: James Wagner

Degree: Master of Science

Title of Thesis: Gibbs Sampling for Gapped Motif Discovery in Proteins

Examining Committee: Dr. Anoop Sarkar, Assistant Professor, Computing Science
Chair

Dr. Martin Ester, Professor, Computing Science
Simon Fraser University
Senior Supervisor

Dr. Fiona Brinkman, Professor, Molecular Biology and Biochemistry
Simon Fraser University
Supervisor

Dr. Arvind Gupta, Professor, Computing Science,
Simon Fraser University
Supervisor

Dr. Cenk Sahinalp, Professor, Computing Science
Simon Fraser University
SFU Examiner

Date Approved: _____

Abstract

Proteins sharing a certain biological role often contain short sequences, or motifs, that are conserved at much greater levels than surrounding areas. The presence of these motifs related to function can be useful in assigning hypothesized functions to proteins in newly sequenced genomes, and, combined with experimental data, help to discern the mechanism of a protein's function. Due to mutations, these motifs will be expected to vary both in length and in amino acid content. Several approaches, including Expectation-Maximization and Gibbs Sampling, have been developed to computationally detect overrepresented motifs in a set of protein sequences.

These approaches have generally focused on the problem of detection of motifs of equal length, and do not work well with certain classes of motifs that do not retain equal length. We provide a novel approach to detection of gapped motifs, which outperforms several traditional motif discovery approaches with several biologically motivated datasets.

Keywords: Gibbs sampling; biological motif discovery; protein subcellular localization; protein classification; overrepresented sequences

Subject headings: Bioinformatics; Computational Biology; Amino acid sequence – Data processing; Statistics

Acknowledgments

I would like to thank my senior supervisor, Martin Ester, for the guidance and support he has provided me at each step of this project. Also thanks goes out to committee member Fiona Brinkman for her guidance and for introducing the subcellular localization problem, and to her graduate student, Nancy Yu, for her invaluable data sets and relevant discussions. My third committee member, Arvind Gupta, also had sage and useful advice whenever I met with him, and for this I am most appreciative. I would also like to extend my gratitude to Martin Ester's postdoctoral fellow Alexander Schönhuth with whom I met regularly and whose discussions were crucial in developing and refining the concepts discussed here.

Funding for my M. Sc. work was provided by the CIHR/MSFHR Training Program in Bioinformatics and NSERC, and I would like to thank all of the faculty and staff affiliated with the Training Program in Bioinformatics for their hard work in seeing that we students are well cared for and well educated. Thanks especially goes out to the program's Coordinator, Sharon Ruschkowski, for her fast and efficient ability to deal with any problems or concerns arising, and my former rotation supervisors Irmtraud Meyer (Computer Science, University of British Columbia), Ryan Brinkman (Medical Genetics, University of British Columbia), and Richard Bruskiewich (International Rice Research Institute, Philippines) for the breadth and scope that I gained in Bioinformatics thanks to their guidance. In an interdisciplinary, multi-faculty, and multi-university program such as the Training Program in Bioinformatics, many people must be involved to ensure that things go as smoothly as possible for the students, and in addition to all of those mentioned above I must also thank the Graduate Program Assistants in the School of Computing Science at Simon Fraser University, Val Galat and Gerdi Snyder, for their attentiveness and conscientiousness regarding the needs of all SFU Computing Science graduate students, myself included.

Thank-you to all of my colleagues in Martin Ester's lab: Rong Ge, Gabor Melli, Richard

Frank, Flavia Moser, Mohsen Jamali, and Recep Colak for a friendly working environment.

My good friends, Rajneil Deo and Sandy Blatchford, provided much personal support and encouragement for the last two years of living in Vancouver and working on my Master of Science degree. Without their concern for my well-being and attentiveness and sympathy for the day-to-day challenges of graduate studies the last two years would no doubt have been much more of a struggle.

I would like to thank my family for the support they provided over their years, and their firm belief in the importance of a good education which they instilled in me from pre-school onwards. This belief was a constant guiding light for me and I would probably not be standing on the cusp of completing a M. Sc. without this belief.

Contents

List of Tables

List of Figures

List of Algorithms

Chapter 1

Introduction

The functional regions of biological molecules, such as protein, DNA, and RNA contain certain regions, or motifs, that are relatively conserved compared to surrounding regions. Focusing particularly on proteins, the molecules that can be thought of as the functional work-horses of the cell, motifs are involved in, or at least correlated with, the various functions of the protein. This could include signals related to localization.

Though certain protein subregions may be indicative of a particular protein-related function, even when seen in proteins from various life forms, they undergo various changes due to mutational events, though perhaps at a lower rate than surrounding areas. Mutations may change individual amino acid symbols within the protein, shift the motif to a completely different area of the same protein, delete existing residues, or insert new residues into the motif region.

Being able to identify such functional regions can be important on several fronts. If one is interested in doing biological experiments on a particular protein in order to understand the mechanism of its function better, having a better idea of subtly conserved regions in proteins performing the same function in distantly related proteins may provide one with a better biological hypothesis of the amino acid residues involved, and, if the function is deleterious, candidate amino acids for mutation or drug targets. One may also build a representation of the motif, in one of various formats, and use this as a classifier against a set of sequences in a database in order to retrieve other candidate proteins containing the same motif and potentially the same function.

Because it is a problem with a much smaller search space, and for many types of motifs

a biologically valid assumption, early research [5, 32] focused on development of motif discovery procedures that took as input a set of proteins believed to share a common motif, and returned motifs that may be at arbitrary locations within the proteins, and may have undergone amino acid substitutions, but did not change in length during evolution.

Known motifs involved in protein subcellular localization of bacteria provide a clear example of the existence and importance of variable length motifs, as do the numerous varying length motifs in databases such as the PROSITE pattern databases [27]. While some reasonably well characterized variable length motifs related to protein subcellular localization have been discovered through experimental knowledge, other forms of localization and types of secretion have as of yet uncharacterized motifs involved, indicating a potential applicability of a variable length motif discovery algorithm.

PRATT [28] is a noteworthy first attempt at discovery of variable length motifs. One major shortcoming is the choice of motif representation used: the regular expression. As will be shown in the discussion of motif representations, this has substantial limitations, and the practical implementation of PRATT falls short in experimental analysis. Neuwald and Liu [39] developed a method for discovery of such motifs, their implementation is not available, and, as indicated in the related work section, has limitations of its own. Parallel to work done in the thesis, Frith et. al. [18] released the program GLAM2. This shares many parallels with the work that we have done related to motif discovery. It makes use of a gapped model very similar to the profile HMM model obtained via Gibbs sampling in our approach, and carries out a Gibbs sampling procedure for determining the optimal motif regions of each protein, though has more sophisticated means of adjusting for optimal motif width than our approach.

Work was also carried out in which the input dataset contained proteins not containing a motif common to the majority of input proteins ("contaminating" proteins). We introduce two methods, graph based subset selection and subsequence clustering, for selecting a subset of proteins whose subsequences contribute to the final motif, and these are compared to GLAM2's means of selecting a subset of proteins, and found to outperform in some cases. Ability to correctly discover and align sets of regular expression containing motifs, and the ability to correctly discovery type II secretion motifs were assessed, with 0%, 20%, and 33% of the input set being non-motif containing ("contaminating") proteins. The main benchmarks used were the set of 58 PROSITE [27] motif protein sets used to test GLAM2, and a set of type II secreted proteins.

Performance in GLAM2 and our approach (Gibbs-HMM) was found to be comparable, though due to their superior means of selecting for motif width, better overall alignments were seen with GLAM2. The ability to better correctly exclude input sequences that do not contain the type II secretion motif was found to be stronger with the extensions to our method than with GLAM2, however GLAM2 had better overall alignment accuracy with the PROSITE datasets when contaminating proteins were present.

As noted, motifs can also be used with the sequence classification problem, and we attempted to do so here, but ultimately found that this falls short with type II secreted motifs. Profile-HMM's have their limitations with respect to representing this type of motif without bringing in an unacceptable level of false positives.

1.1 How this document is organized

Chapter 2 will review subcellular localization: its biology, motifs involved, and computational prediction. This served as the original motivation for motif discovery. Chapter 3 discusses the representations of motifs, and existing motif discovery work. Chapter 4 will present an outline of the basic Gibbs-HMM method that we make use of, as well as two extensions to the basic approach designed to address particular cases. Chapter 5 will present the results of experimental approaches. Chapter 6 lists several future directions for investigation and research.

1.2 Contributions of this work

This works demonstrates the adaptability of the Gibbs sampling method for gapped motif discovery. At the time of the development of most of the work, it was the only known research involving Gibbs sampling with a profile-HMM motif representation. GLAM2 was recently published and also found to involve a motif representation very similar to a profile HMM and also uses Gibbs sampling to determine the parameters of this model. Our methods diverge considerably in terms of how proteins are selected or excluded for usage in the model, and in some cases our methods were found to outperform those of the state-of-the-art GLAM2 motif discovery method.

Chapter 2

Subcellular localization

2.1 Biological background and localization

The bacterial cell consists of the cytoplasm, surrounded by a lipid bilayer called the cytoplasmic, or inner membrane. Gram negative bacteria have an additional lipid bilayer beyond the inner membrane called the outer membrane, as well as a space between the two membranes referred to as the periplasm; Gram negative bacteria also possess a thin cell wall which is not regarded in most prediction methods as a major target of localization. Gram positive bacteria are further surrounded by a thick peptidoglycan cell wall, in lieu of an outer membrane and periplasmic space.

Proteins can be thought of as the functional workhorses of any type of cell, including bacterial cells, with functions ranging from metabolism to motility. In the simplest possible encoding, proteins can be regarded as 1-dimensional strings of amino acids, having a possible 20 letter alphabet.

Focusing on the Gram negative example, of which more literature regarding localization abounds, following the synthesis of proteins in the cytoplasm, a protein can be targeted to one or more of these subcellular compartments, as well as the extracellular space (or other cells) surrounding the cell. A number of mechanisms have been discovered experimentally that carry out this targeting of proteins to the different localizations. Many proteins possess an N-terminal sequence motif that enables the protein to be translocated, via the general secretory (Sec) [44] pathway through the cytoplasmic membrane. Proteins translocated in this fashion may remain in the cytoplasmic membrane or periplasm, or be transported to the outer membrane or extracellular space via other methods. Another pathway with

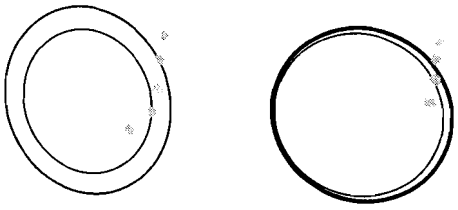

Figure 2.1: Bacterial protein subcellular localization on the left is the Gram negative morphology of bacteria, with dots indicating the five major localizations from inside to outside: cytoplasmic, cytoplasmic (inner) membrane, periplasmic space, outer membrane, and extracellular space. On the right is the Gram positive morphology, with dots indicating the four major localizations from inside to outside: cytoplasmic, cytoplasmic membrane, cell wall, and extracellular space.

well-characterized motifs is the Twin Arginine Targeting (TAT) [37] pathway which also translocates proteins across the cytoplasmic membrane, into the periplasmic space, where most of them remain. The outer membrane insertion pathway is a poorly characterized pathway that pulls proteins from the periplasmic space to the outer membrane. Described in the following paragraphs are means by which a protein may be secreted all the way to the extracellular space of a bacterium.

The gram-negative type I secretion system is carried out by a translocator consisting of three proteins that span the inner and outer membranes, bypassing the periplasm and shuttling proteins directly to the extracellular space [14]. The type II secretion system, [44] common to both Gram positive and Gram negative bacteria is a two-step process in which proteins are first translocated across the inner membrane by either the Sec of Tat [37] pathway. It is possible for proteins to remain in either the inner membrane or periplasmic space at this stage, in which case a protein cannot be said to be truly type II secreted. In a separate stage proteins are then translocated across the outer membrane into the extracellular space via an outer membrane apparatus of 12-16 proteins called the secreton. [12]. Type III secretion systems are classically injectisomes, apparatuses that commonly allow bacteria to inject proteins directly into a eukaryotic host cell via a complex structure of proteins spanning the two membranes. These injectosomes share an ancestral relationship with the flagellar apparatus that plays an important role in motility [30].

Type IV secretion systems are involved in the secretion of toxins and bacterial pathogens into host eukaryotic cells and other bacteria. They consist of a complex apparatus spanning both membranes that can secrete protein and DNA. [11].

There are also several self-sufficient pathways capable of exporting proteins across the outer membrane without the use of ATP, following export via the Sec or Tat pathways. These include autotransporter 1, autotransporter 2, two partner secretion, and chaperone/usher pathways. [52]

2.2 Subcellular localization prediction

Experimental determination of where inside, or outside, of a cell a protein localizes remains a labour intensive process that cannot nearly keep up with the rapid rate of bacterial genome sequencing. This leads to the question of whether a protein's subcellular localization can be computationally predicted, given only its sequence.

There are a number of reasons that the determination of a protein's subcellular localization is of interest. A protein's localization can provide an important clue as to its possible function. Also, as a genome of a bacterial pathogen is newly sequenced, many in the pharmaceutical industry would be interested to determine proteins located towards the outside of the cell (outer membrane or extracellular space) as these would be more likely candidates for drug targets. Localization may also provide interesting clues to bacterial evolution, namely how the distribution of protein localization changes based on the environment and needs of different species or strains.

A number of methods for computational prediction of protein subcellular localization have been developed in recent years, for a more extensive review see [21]. In all methods described the protein is encoded as a 1-dimensional string, corresponding to the sequence of its amino acids. No other prior information about the protein, which cannot be computationally inferred or predicted from the primary amino acid, sequence is used.

PSORTb [19], [20] involves a series of independent modules, each making a prediction of a particular localization, or, if not enough certainty is present, returning a value of Unknown Localization. The predictions from the various modules are integrated via a Bayesian network, which outputs a predicted localization for each protein, or Unknown if no individual module gave a prediction, or too much conflict is present between the modules. Viewed systematically, the various modules are indicative of many of the general strategies that have

been used in computational approaches for the prediction of subcellular localization. There is a BLAST [3] module in which a database of proteins of known localization is searched, with the protein of unknown localization as the query. The localization of the protein giving the best alignment to the query protein (where best alignment is based indirectly on the edit distance) is then the predicted localization for the query protein. A series of support vector machines (SVM's) are also used. SVM's are a discriminative method that serve best for two class problems, but do not require the data to be linearly separable. A series of items of known class label are input to an algorithm that determines the maximum separation possible between the two classes. This maximum separation is actually a hyperplane. Data points of unknown class label can then be scored against this hyperplane and a class label assigned accordingly. Typically, support vector machines require that each data point input for either training or for scoring be a vector of real values. Since proteins are here encoded as 1-dimensional strings, of variable length, a transformation is required. In the case of PSORTb, frequent subsequences within a set of proteins of common localization are determined. The values within a protein's vector are then set to 1 indicating the presence of this frequent subsequence, and 0 otherwise.

Proteome Analyst [35] is one method that diverges somewhat from what is commonly seen amongst prediction methods. Like one module of PSORTb, it carries out a BLAST search, however the database search consists of the full SWISS-PROT database, including proteins of unknown localization. Keywords are extracted from the top protein hits, and are then passed to a Nave Bayes classifier to assign the protein to one or more localizations.

The latest version of Cello [56] makes use of support vector machines. They employ a number of strategies for extracting numeric information from a protein that can be input into a support vector machine, including amino acid composition, n-peptide composition, and gapped dipeptide composition.

PSLPred [9] is another tool that makes use of support vector machines, in addition to values based upon the amino acid composition of the protein, certain real-valued physicochemical properties of the protein are determined based on calculated values for the individual amino acids, which are then used as values in the vector used to train and score the SVM.

PSL101 [51] shares a common linkage with PSORTb in that it makes use of a variety of modules providing different biologically motivated insights regarding localization prediction. In addition to an SVM based upon many of the same features as described with Cello

above, as well as features based on solvent accessibility and secondary structure, they also apply computational methods for the detection of certain sequence features associated with particular localizations. These sequences include the Sec motif for translocation across the inner membrane, transmembrane alpha-helices, TAT motifs, and transmembrane beta-barrels (for outer membrane proteins).

If one considers the various methods for computational prediction of bacterial protein subcellular localization, several limitations come to mind. Considering first SVM-based methods, support vector machines are an excellent discriminatory tool, and solid attempts have been made to extract biologically relevant features of a protein, such as amino acid composition and physico-chemical properties, for use in an SVM. However, support vector machines are weak when it comes to interpretability. One is presented only with a localization prediction, with little ability to analyze what features of the protein in particular led to this prediction, or means of weeding out a potentially spurious prediction. Sequence alignment based methods rank better in terms of interpretability, as one can readily see which proteins a query protein gave a significant alignment to, but these too have their drawbacks. One obvious one is the dependence on a comprehensive database of proteins of known localization, or, in the case of Proteome Analyst's keyword based approach, proteins with keywords relevant to localization. Perhaps of greater methodological importance is the fact that, using local alignment methods such as BLAST, proteins may give a significant alignment while not sharing a common localization. In this case, the alignment is based upon significant similarity within a functional domain unrelated to localization. PSORTb addresses this issue by requiring that the region of significant alignment be at least 80% of the length of both the query protein of unknown localization, and the hit protein of known localization, thus effectively rendering the alignment tool a global one. The downside of this is that proteins are typically not believed to localize to a particular site based upon global sequence characteristics, but rather particular short motifs are mainly responsible. Thus scoring a query protein for the presence or absence of these motifs would seem like the ideal solution. PSORTb already makes use of many of these motifs towards localization. Often these motifs are not the true motifs responsible for the protein's localization, but are still nevertheless very useful clues, as all other proteins of known localization that share the particular motif also share a common localization (100% precision). Given our present knowledge about most means of localization, and the motifs of target proteins involved, this is a perfectly acceptable compromise. When considering PSORTb as a system, most protein

predictions are made based upon BLAST hits, or the support vector machines. For all of their pitfalls, these tools still achieve better overall coverage, with reasonable precision, than the tools based upon motifs. Thus, a reasonable goal in trying to enhance both the interpretability, and the overall coverage of a subcellular prediction program such as PSORTb would be to add to the set of localization-related motifs against which query proteins are scored.

2.3 Known motifs relevant to localization

In what is regarded as the "classical" form of protein translocation, the Sec-dependent pathway of translocation across the cytoplasmic membrane, the signal motif involved does not have a concrete consensus sequence, but rather a three region structure composed of the n-, h- and c-regions. The most amino-terminal n-region often contains positively charged residues. The central h-region is a region of about 7 to 15 predominantly hydrophobic residues, followed by a more polar, carboxy terminal c-region [38]. A variety of tools exist to computationally determine whether to not a protein sequence has the N-terminal Sec motif, most notable is SignalP [41], which combines neural network and Hidden Markov Model based predictions. Motifs involved in the Main Terminal Branch of the type II secretion pathway, responsible for proteins being translocated across the outer membrane, are less well characterized.

The Twin Arginine Translocation pathway is another pathway able to translocate proteins across the cytoplasmic membrane. In this case, proteins display a consensus motif of S-R-R-x-F-L-K [37]. Tat signal peptides are another example of a motif for which computational tools already exist, namely TatP [8].

Many, but not all of type I secreted proteins characterized so far have a multiple glycine rich repeat regions that specifically bind calcium ions. This calcium binding region should be clearly distinguished with what is also termed the C-terminal signal secretion signal, also a well-conserved region consisting of negatively charged residues followed by three hydrophobic residues [14].

Type III secretion is believed to rely on a structural signal with little primary sequence conservation in the N-terminal region. This signal remains poorly characterized [30]. Recent in-vivo studies have suggested that a C-terminal region is necessary for type-IV dependent secretion [11]. A consensus sequence among particular Agrobacterium tumefaciens secreted

proteins has even been developed: R-X(7)-R-X-R-X-R-X-X(n) [1].

Amongst the various types of proteins that can be translocated across the Gram negative outer membrane in the absence of ATP, the C-terminal translocator domain seen in auto-transporters is a well conserved region of 250-300 residues in length predicted to fold as 12-14 stranded Beta-barrels. These already exist as recognized domains in the Pfam database. Proteins secreted via the two-partner secretion pathway also have a well conserved N-terminal domain [52].

Though the pathway for insertion of proteins into the outer membrane remains poorly understood, many proteins inserted in the outer membrane share common structural characteristics, namely Beta-barrels. Zhai and Saier [57]. developed a computational tool for finding beta-barrel proteins, and applied it to the Escherichia coli genome.

Gardy et. al [19] have identified a number of PROSITE [27] patterns that, while not directly responsible for localization, are still associated with one particular form of localization, with 100% precision. They also applied a frequent subsequence based approach to identify motifs specific to the outer membrane.

When taking stock of the current state of knowledge regarding motifs related to localization one can see there is still room for improvement. While identification of proteins that cross the inner membrane using the Sec or Tat pathways is a well-developed field, current knowledge of motifs that are involved in subsequent translocation across the outer membrane into the extracellular medium is still in its infancy. Some headway has also been made in detecting particular species-specific motifs involved in type IV secretion, and less progress still with regards to type III secretion motifs. Most mispredictions currently made by PSORTb fall under one of two categories: cytoplasmic proteins wrongly predicted as cytoplasmic membrane, and vice versa, and extracellular, outer membrane or periplasmic proteins being wrongly predicted to one of these other three categories. Thus motif discovery for bacterial protein localization should focus on improving discrimination particularly between these categories.

Chapter 3

Sequence motifs

3.1 Introduction

Sequence motifs are regions of biological sequences that can describe and identify features in DNA, RNA, and protein sequences. The features and functions of the sequence motifs can vary widely. In DNA, the most commonly studied motifs include transcription factor binding sites [48]. Typically located upstream of protein coding regions, these are regions to which transcription factor proteins bind to either facilitate or block transcription of the coding region into RNA, which, if transcribed, is subsequently translated into functional proteins.

Protein motifs may often perform important signaling functions in the cell, through the interaction with other proteins. Using the example of N-terminal signal sequences in Type II secretion [44], this 18-30 amino acid region, present in the target protein to be secreted, interacts with the SecB chaperone protein as the target protein is still being translated and is an unfolded state. The SecB chaperone targets the protein to the membrane bound component of the translocation machinery.

In their recent paper on the identification of biological motifs, Frith et. al. [18] identify three somewhat distinctive classes of biological motifs. The first includes short motifs found at functional sites such as cleavage and binding sites. The second comprises longer motifs associated with structural domains in globular proteins. The third is composed of recurring motifs arising from evolutionarily recent duplications of DNA sequences. Secretion signals would probably fall under the first category, as they are not large structural regions responsible for the protein's actual function, but shorter, harder to identify regions that interact

with other proteins that help to guide a protein to its final localization.

As Frith et. al. point out [18], one single continuous region of protein may not be always be sufficient to explain a biological function, and a function may in fact involve the interaction of multiple dispersed regions brought closer together only due to the complex three dimensional structures of proteins. At the very least, however, motifs can still be correlated, if not explanatory for biological functions, which can still be useful in annotating the possible function of a molecule.

A number of databases of known motifs have already been compiled including ELM [45], PROSITE [27], BLOCKS [24], and PRINTS [4]. Longer protein domain databases with profile-HMM based representations include PFAM [6] and TIGRFAMs [23]. PSORTb's [19] classification system already makes use of numerous motifs in which all proteins of known localization containing the motif are associated with only a single localization (100% precision). Even without a solid biological explanation for the reason a motif is associated with a particular function or localization, motifs can still serve as a useful predictive tool.

The starting point for any representation of a motif is usually an alignment of a set of proteins, or multiple alignment. The means of obtaining a multiple alignment can vary greatly in terms of the evolutionary assumptions, degree of manual intervention, and methods used, and a comprehensive review of such techniques is beyond the scope of this thesis. The input to a multiple alignment procedure would be a set of proteins or protein regions believed to have some sort of evolutionary relationship. Though full proteins may be globally aligned, this serves as a much more difficult problem as related proteins may have large domains added or deleted over the course of evolution while still retaining strong levels of conservation at particular domains or regions of interest. Certainly for a motif representation the alignment involves protein subregions showing higher levels of conservation than seen in surrounding areas.

In a multiple alignment, the set of protein regions is given, with each protein being a row. Ideally, each column will be a set of homologous amino acid residues, with homologous referring to the belief that, though the residues may be different, they have diverged from the same residue in a common ancestral protein. In the case of ungapped multiple alignments, only amino acid residues will be present in any column. In gapped multiple alignments, columns may include gaps, representing cases where either the ancestral residue in particular proteins has been deleted, or cases where non-ancestral amino acid residues have been added to a subset of the proteins, with those proteins that have not undergone such insertions

```
Z3H7A_HUMAN/859-881      Cwm..CgknCnsekqwqgHissekH
Z512B_HUMAN/142-163      Cpf..CeaaFtsktqlekHriwn.H
Z512B_HUMAN/542-563      Cqh..CrkqFkskaglnyHtmae.H
Z512B_HUMAN/632-653      Cth..CgktYrskaghdyHvrse.H
Z512B_HUMAN/786-807      Cll..CpkeFssesgvkyHilkt.H
Z518A_HUMAN/236-256      Cgk..ChhvCftkgelqkHlhi..H
Z518A_HUMAN/1451-1471    Cwf..CgrvFdnqdtwagHgqr..H
Z518B_HUMAN/1038-1058    Cwf..CgrlYedqeewmsHgqr..H
Z561_ENCCU/222-244       CpyegCtmeLptlsrikrHyiv..H
Z561_ENCCU/252-274       ClnkdCnkrFsrkdnmlqHyki..H
```

Figure 3.1: Multiple sequence alignment of several zinc finger motifs, adapted from the PROSITE website [43]. Cases in which insertions have occurred are indicated in columns where the majority of proteins have a "." symbol, indicating a gap, with a minority of proteins having an actual amino acid symbol for that column.

having gaps added in the columns corresponding to inserted residues.

Two proteins from different organisms can be said to share a certain motif even if one protein contains no subregions that are identical to subregions of the other protein. Due to mutations that occur during the course of evolution, it would indeed be surprising if one did see two continuous, identical regions in the two proteins. Rather, the motif regions of the two proteins would be expected to have a higher level of similarity, with a higher proportion of identical or similar amino acids, than seen in surrounding areas. Mutations may result in substitution of amino acids for other amino acids, or the insertion or deletion of amino acids, or the movement of the motif to a different region of the protein and yet the function, if any, associated with the motif remains. Thus, motifs cannot be represented simply as strings with any hope of finding anything but a small fraction of sequences actually containing the motif. More complicated representations, allowing for degeneracy, are necessary. We will turn now to ways in which motifs are actually represented in practice, with reference to several protein motif databases.

3.2 Protein Motif Representations and Databases

3.2.1 Regular Expressions

Beyond the realm of a simple string representation, perhaps the simplest representation possible of a motif is the regular expression or pattern. From 1988, curators of the PROSITE

[27] database have manually developed a large number of these patterns from proteins in the literature. Typically, curators identify groups of related proteins from review articles published in the literature, then, focusing on regions known to be of biological importance, curators manually scan regions of proteins looking for short, conserved regions that can be the starting point for a pattern. Biological patterns are represented in a similar fashion to what one may see in a regular expression syntax of formal language theory, although the PROSITE developers do not allow for the full set of expressiveness one sees in regular expressions in formal language theory. For each position within a regular expression, one may have either a standard one letter amino acid code, a set of one letter amino acids to indicate the set of possible amino acids, eg [A G S] to indicate a choice between the amino acids Alanine, Glycine, or Serine for that position, or an x indicating that any amino acid may be present at that position. Repetition of the degenerate letter 'x' is permitted, for example x(3) corresponds to exactly 3 instances of any amino acid, while x(2,4) corresponds to at least 2 but no more than 4 instances of any amino acid. Ranges are only permitted with x, for example A(2,4) is not a valid pattern element. Regions of arbitrary length, though allowed in standard regular expressions, are not allowed in PROSITE regular expressions, for example there is no valid expression corresponding to x*, 0 or more repetitions of any amino acid.

An example regular expression used to query a database of proteins could be [AC]-x-V-x(2,4)-[ML].

A database sequence matching this query regular expression could be TGNARVANCMDECG. Because these are short regular expressions scanned against whole proteins, there may be flanking regions on either side of the regular expression. In this case TGN is outside of the regular expression. A matches the first position of the regular expression, which, following the [AC] element may be an A or C. R corresponds to the x, which may be any amino acid. Next we have a match to the only acceptable amino acid for the third position, V. Next we have between 2 and 4 of any symbol from the alphabet. ANC fits this criteria. Then there is an M, corresponding to the M or L in the last position of the regular expression. finally the DECG comprises the flanking (but not necessary) region of the sequence.

Regular expressions are a completely qualitative form of assessing motifs. A sequence either matches or does not match a regular expression, without a score assigned to it. The above example matches it. If we were to have a database protein of TGNTRVANCMDECG,

this would no longer match, as the first position of the regular expression must be an A or a C, and we have now substituted the corresponding position in the database protein with a T. If one had strong reason to believe this protein region was a valid matching instance of the biological function represented by the regular expression, one would either have to accept that this case is excluded by the expression (false negative), or add T to the list of choices for the first position, with no way of indicating the fact that the T is rarely seen compared to A or C. This increased degeneracy may bring in false positive proteins that match the more relaxed regular expression but are unrelated in terms of function.

A number of advantages exist for regular expressions over the more complicated representations which follow. They are easily intelligible for the user and also point the user towards the most conserved residues, which are also the most important for function. A large set of proteins can also be scanned for the presence or absence of a set of patterns in reasonable time [49].

3.2.2 Position Specific Scoring Matrices

There are inherent limitations in the usefulness of regular expressions in the identification of more distant motifs. Regular expressions do not accept any mismatches, and hence for cases where certain substitutions are rare but still happen, one must either completely rule out sequences that have this rare substitution, or allow for an ever increasing number of amino acids at that column, thus making the expression too degenerate to be of practical use in certain cases.

This problem is alleviated by the use of profiles, which are quantitative means of representing motifs. For each column of a motif, they consist, not just of a set of possible amino acids, but a probability vector of length 20, with a probability for each amino acid present at that column. A mismatch at a highly conserved position can still be accepted provided the rest of the motif displays a sufficiently high level of similarity.

The BLOCKS database [24] makes use of position specific scoring matrices (PSSM's) to represent motifs against which sequences can be scored. In this implementation, ungapped multiple alignments for groups of related proteins are obtained by means of an ungapped motif discovery method. For each column, counts are made of each of the 20 amino acid residues. In the very simplest form this vector of 20 probabilities could be used as the set of probabilities for that column. In practice they are converted to a set of log odds scores. The probability of seeing an amino acid in a column is divided by the expected frequency of

that amino acid in a random sequence, which can be readily derived from a large database of proteins. The logarithm of this ratio is then taken due to the fact that multiplying long lists of these ratios results in very small numbers. The summation of these logarithms is then equivalent to the multiplication of the actual ratios. For an amino acid a, if we have the probability p_a of this amino acid appearing at a site in a random protein sequence, n_{ac} counts of this amino acid in column c, and n_c total amino acids in column c, the score s_{ac} for amino acid a in column c would be:

$$\log \frac{n_{ac}/n_c}{p_a} \tag{3.1}$$

In practice, taking only the counts of observed amino acids is not a good strategy. Particularly in cases where the alignment is of only a small set of protein regions, there will be cases of columns with 0 counts for particular amino acids. This would obviously create an impossible situation for taking the log-score, and even if we were not to, matching sequences having an amino acid symbol in a certain column unseen in the training sequences would end up with an unnecessary probability score of 0. Pseudocounts therefore are added to the observed counts (n_{ac}) seen for the amino acids in each of the columns. In the simplest case we could use Laplace's rule and add one to every amino acid frequency in every column. However better estimates of the best pseudocounts to add, based upon the observed amino acid counts, can be made use. Henikoff and Henikoff [26] review several, perhaps the most sophisticated being the Dirichlet mixture method.

This is the method used to add pseudocounts to the PSSM's used during MEME's motif discovery method [5], as well as the HMMER based profile-HMM's [16] that will be discussed shortly.

The Dirichlet distribution can be thought of the distribution over a set of multinomial distributions. In turn, a multinomial distribution governs the actual counts of a particular set of elements observed. In this case, for a particular column of a multiple sequence alignment, the observed counts of amino acids are governed by a multinomial distribution, which was sampled from a Dirichlet distribution. Or, alternately a certain fair or unfair dice will have an underlying multinomial distribution governing the counts of 1, 2, 3, 4, 5, and 6, seen in a set of rolls, and, if there are variations in the dice, the dice factory samples from a Dirichlet distribution to determine the multinomial distribution associated with a dice [15].

Proteins are complicated enough that not only one Dirichlet distribution can govern all amino acid distributions, in certain chemical or physical environments one can expect to see

different distributions of the amino acids, and a mixture of Dirichlets is more appropriate.

How does this relate to pseudocounts? If we have an observed set of amino acid counts for the column of a multiple sequence alignment, and a mixture of Dirichlet distributions we can obtain the probability for each Dirichlet distribution given the observed conts of amino acids. Using these probabilities as weights, a pseudocount for each amino acid, from each of the mixture models involved is added to the count vectors.

By whatever means of obtaining a reasonable log odds score, once we have a PSSM, the PSSM can then be slid along a candidate sequence and at each alignment the score for every amino acid in the sequence is looked up in the column of the PSSM with which it is aligned. Then the scores for all of the columns are added to arrive at the alignment score. The maximum of all alignment scores taken over the course of the sequence is then assigned as the score for the sequence.

In making use of fixed length, ungapped alignments resulting in a fixed length model the position specific scoring matrix is still a relatively simple form of motif representation to obtain. As the review of motif discovery algorithms will indicate, discovery of ungapped motifs in protein sequences is a fairly mature field. Perhaps the most challenging part comes with obtaining good pseudocounts to prevent the PSSM's from being overfit to the set of training sequences, and we have enough known examples of protein substitutions to develop some useful guidelines for obtaining pseudocounts.

For many types of protein domains the assumption that there will be no insertion or deletion of amino acid residues is a valid one, and the PSSM is a reasonable representation of the region. In other cases there will be insertions or deletions, and a protein with the deletion or insertion of even a single amino acid within the motif region would receive an unnecessarily low score when scored against the PSSM. We now turn our attention to two related forms of representation that are quantitative and allow for insertions and deletions.

3.2.3 Generalized Profiles

In addition to a large selection of regular expressions, the PROSITE database makes use of Generalized profiles to represent proteins with gaps, an extension of Gribskov et. al's [22] methods. As usual, a multiple alignment of protein regions is used as initial input. Log-odds scores for the twenty amino acids in each of the columns are computed as we would for standard PSSM's. In addition, a 21st column is added to represent the penalty given to the score when a gap is encountered at that position. Due to the opportunity of

encountering gaps, The process of aligning a protein to a generalied profile requires a bit more computing than the sliding window scanning process that is used with PSSM's. In this case a dynamic programming based alignment, used very commonly with standard sequence alignment systems [50] can be easily adapted to the task.

In justifying their choice of using generalized profiles to represent motifs in the PROSITE database, Sigrist et. al [49] point out important advantages and disadvantages of generalized profiles compared to profile HMM's, the final motif representation whose in-depth discussion follows. The main advantage of generalized profiles is their ease of manipulation. Because profile HMM's are a generative model, the sum of the probabilities of all possible sequences being generated by a profile HMM must sum to one. One cannot increase the probability of one sequence being generated without decreasing the probability of another sequence. Generalized profiles are not generative models for sequences and one can easily manipulate scores for a generalized profile in a text editor. In contrast, profile HMM's are formally built upon probability theory.

3.2.4 Profile HMMs

Profile Hidden Markov Models (Profile HMM's) are statistical models of multiple sequence alignments. They combine the statistical framework of Hidden Markov Models (and all of the algorithms that can be effectively used with HMM's) with the consensus representation of a multiple alignment in the form of a profile.

An HMM essentially describes a probability distribution over a potentially infinite number of sequences. An HMM passes through a series of states, and, restricting the scope of our discussion to discrete cases, the states may be silent, or emit symbols taken from a discrete alphabet. For each non-silent state, there will be a set of emission probabilities distributed over the alphabet that govern the probabilities of that state emitting each of the symbols of the alphabet. For each state, an HMM will also specify a probability distribution of transitions to all states in the model, known as the transition probabilities.

The extension of HMM's to Profile HMM's is a straightforward and intuitive one. The sequences that are emitted are biological sequences (DNA, or, more commonly, protein); typically the set of sequences used to construct a common model are believed to share some sort of evolutionary relationship such that it makes sense to construct a single model from them. The alphabet of symbols would be the twenty letter amino acid alphabet for proteins, or the four letter nucleotide alphabet for DNA or RNA. When the sequences are multiply

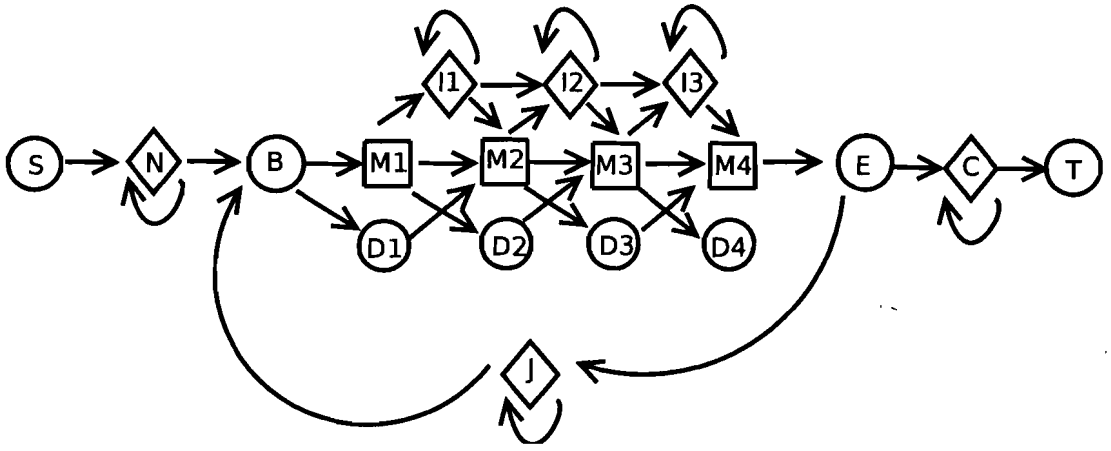

Figure 3.2: HMMER Plan7 architecture used throughout this work, adapted from [16] by permission

aligned the resulting alignment can be visualized as a matrix with the rows corresponding to the input sequences and the columns consisting of either symbols drawn from the alphabet or gaps. If the sequences used in the multiple alignment are assumed to all be variants of a common, ancestral sequence one can view the gaps as either deletions (i.e. the corresponding column exists in the ancestral sequence but a particular sequence, or row, has undergone a deletion for this column) or insertions (i.e. the corresponding column does not exist in the ancestral sequence but a particular sequence or sequences have undergone a the insertion of a new column, while those sequences that have retained the ancestral structure and not undergone an insertion will have gaps as entries for the column). If a symbol from the alphabet is entered at a particular column, it is regarded as a match to the ancestral sequence at that column, though very often a mutation to another symbol of the alphabet has occurred.

The matches, insertions, and deletions can be translated to hidden states in a profile HMM model, with each column of the original alignment that corresponds to the the ancestral sequence (that is, columns that are not insertions into the original sequence) having a corresponding match, insert, and delete state. Match states will emit symbols from the alphabet. Those columns that represent insertions are not represented as separate columns in the model, but transitions to the Insert state of the most recent column that is part

of the ancestral sequence. In a multiple sequence alignment in which there are x ancestral columns, there will be 3x match, insert, and delete states, with one of each for each ancestral column. Arbitrary transitions between any of these 3x states are not permitted. Practical considerations also further restrict the architecture in actual implementations of profile-HMM theory. The profile-HMM architecture used throughout this work is the plan 7 architecture used in the open source profile-HMM software HMMER.

The I, M, and D states correspond to the insert, match, and delete states for the columns of the multiple sequence alignment. As can be seen there is a linear ordering in terms of the transitions allowed. Often multiple amino acids may be inserted in a location that corresponds to only a single symbol in the original ancestral sequence, this necessitates the loops from Insert states to themselves, with one symbol being emitted for each transition to the insert state. Since each column corresponds to a single symbol in the original ancestral sequence, there can only be one match state per column, and one only symbol that can be deleted, hence the absence of loops for these two types of states.

S and T states are silent states corresponding to the start and end of the model. Since in practice profile HMM's are aligned to full protein or DNA sequences, of which only a small portion correspond to the actual domain one also needs N and C states, referring to, in the terminology of protein chemistry, the N-terminal and C-terminal regions that lie upstream and downstream of the actual domain represented by the I, D, and M states of the model (referred to as the main model by the developers of HMMER). These states will be responsible for emitting the symbols that lie before and after the symbols emitted by main model states. After passing through the N state corresponding to non-domain symbols, the model will pass through a non-emitting B state representing the beginning of the main model. Similarly, the model passes through a non-emitting E state immediately after exiting the main model section. In cases in which there are multiple instances of a domain in a single protein or DNA sequence, looping through the J state will allow for emission of symbols that lie after the end of one domain region represented by the main model, and before the beginning of the next domain region.

Constructing a profile-HMM

Eddy [16] differentiates between two ways of constructing a profile-HMM. Training a profile-HMM consists of inputting a set of sequences in which the multiple alignment is not known. If the multiple alignment is not known for a particular sequence, then the state path through

M, D, and I states will also be unknown. In this case traditional HMM algorithms such as Baum Welch must be used which will perform simultaneously the multiple alignment of the sequences and the setting of parameters for the transition and emission probabilities. On the other hand, building a profile-HMM will take as input a set of sequences that have been multiply aligned, in which the alphabet symbols and gaps have already been assigned to columns, and setting the emission and transition probabilities is a simpler case of counting symbols per column as well as transitions between states, adding appropriate pseudocounts, and converting these to appropriate emission and transition probabilities (HMMER also carries out a maximum a posteriori algorithm to appropriately assign columns of the multiple alignment to either columns that were part of the original ancestral sequence, or inserted columns). The HMMER developers [16] cite the local optima that optimization algorithms for training a profile HMM from unaligned data can run into as the reason for HMMER, in its current implementation, accepting as input only pre-aligned sequence data, and building a profile HMM. It is not intuitively obvious, however, why a multiple sequence alignment algorithm cannot fall into local optima of its own.

Columns of the alignment that correspond to Match columns, that is residues that have a corresponding residue in the ancestral protein sequence, are identified using a maximum a posteriori method, other columns are treated as insertion columns. Amino acid emission probabilities for Match states are then used in a very similar manner to what is seen with the columns of a position specific scoring matrix. Amino acids are counted, then Dirichlet mixture-derived pseudocounts are added for each amino acid. The probabilities are divided by null probabilities of amino acids seen in random protein sequences, and the log is taken to obtain scores for each amino acid in each column. Deletions in the ancestral, match columns are counted independently for each column, as are insertions of amino acids into non-match columns and from these transitions probabilities to insertion and deletion states are obtained. Because the multiple alignment seen in the set of proteins available for the model may over- or under-represent the true amount of insertions and deletions seen in all possible proteins (known and unknown), Dirichlet priors (only a single component) are also used for the transition probabilities, these were based upon large scale alignments of protein domains. Insertion emission probabilities also use priors different from those seen with match emission probabilities, more reflective of the types of amino acids seen in known insertions.

Relevant HMM-related algorithms

Given a profile-HMM (or any other type of HMM), there are several questions that we can ask, and corresponding algorithms that do so.

Viterbi

Given a sequence, what is the most likely path through the HMM (and what is the probability of this path)?

In our case, we are presented with a sequence of amino acids, and a profile-HMM. We wish to know the most probable state path (which amino acids correspond to match or insert emissions? Does it pass through any delete states?). In essence, we are determining the most likely alignment of the amino acid sequence to the profile-HMM model.

Viterbi makes use of a dynamic programming approach, in which, starting from the first element of the sequence, for each possible state the maximum possible score (or probability) obtained from any state path that ends in the state at the current sequence element is stored. The matrix continues to be filled out, with the maximum score for each state ending at each sequence element stored in the matrix, until the last sequence element is reached.

The most probable path can be found recursively. As it turns out, the probability of the most probable path is a useful approximation to the overall probability of a sequence given a model. This is the default implementation used by HMMER whenever one wants to obtain the probability of a protein sequence using a model, and is used throughout here as well.

Forward Given a sequence, what is the overall probability of this sequence passing through the profile-HMM model?

One again, we are presented with a sequence of amino acids, and a profile-HMM. This time, the probability of all possible paths (all possible alignments) is obtained.

This is done with the same dynamic programming based approach as seen in Viterbi, however for each state and each sequence element, the sum of all previous alignments up to the sequence element before the current sequence element is stored, rather than only the maximum score.

Baum Welch

This is an EM based approach to learning the emission and transition probabilities, given the emitted sequences only.

In this case the emitted sequences would correspond to a set of unaligned motif sequences. The iterative, EM approach involves, in words, calculating the probabilities of state paths given the current model parameters, then optimizing the model parameters given the probabilities of the state paths.

In principle this can be used, in practice HMMER has discontinued the implementation of the Baum-Welch approach for training a profile-HMM, favouring instead the approach of building a profile-HMM from pre-aligned sequences. As discussed earlier, their reasoning is that the EM approach is more prone to local optima, though how this is avoided by pre-aligning the sequences is not obvious.

3.2.5 Conclusion to Motif Representation

One of the first, if not the first, protein motif databases was PROSITE, which began in 1988 as a set of manually derived regular expressions from proteins in which the domains involved in a certain common biological function were well characterized by biological experimentation. With the increasing proliferation of biological sequence data (DNA and protein sequences) in the 1990's, came an interest in finding motifs in sets of related sequences in a more automated fashion. An input set of proteins would be known to share some common function, localization, or other biological commonality, but the motif responsible is too short and located at such varying places within each protein that the motif regions do not align when a multiple alignment of the full proteins is carried out. We will now turn our attention towards several of the more prominent protein motif discovery algorithms.

3.3 Motif Discovery Procedures

3.3.1 Pratt

Pratt [28] is a tool for discovery of regular expressions within set of unaligned proteins. It makes use of a minimum threshold of sequences that any output regular expression must match. In the basic algorithm, Pratt starts with the empty pattern (e), which matches all sequences. The algorithm then branches out to all regular expressions (using the restricted definition of regular expressions discussed for PROSITE patterns). All possible extensions of the empty pattern that involve adding either a single amino acid or a restricted set of amino acids is then considered. Those extensions that result in a regular expression not matching

the minimum threshold of sequences are not further considered. In subsequent iterations extensions that involve adding a fixed length wildcard (series of x) followed by a single amino acid or set of amino acids are also considered. Pratt also makes use of several more advanced techniques for extending the regular expression using flexible length, as opposed to fixed length wildcard regions without blowing up the search space. The bottom line is that regular expressions, while simple and suitable for certain simple types of motifs do not have the expressive power for other classes of motifs, such as the localization motifs we are interested in.

3.3.2 MEME

In the expectation maximization (EM) approach [33], the motif is encoded as a position specific scoring matrix, which is refined through a series of alternating E and M steps. The input provided is once again a set of unaligned protein or DNA sequences, and the unknown information that is to be found are the start sites of the motif (of fixed length) which can then be easily aligned to build a position weight matrix. At each E step, for each sequence, the probability of seeing the sequence given each possible start site is calculated. Now, using the probabilities that the site starts in each of the possible position as weights, a summation of each of the entries in the position weight matrix is done. For example, if the probability that a window starts at position 50 of the third sequence is 0.01, and position 50 of the third sequence is an A, then 0.01 will be added to the counts used in the PSSM entry for row A in the fiftieth column. The maximization step is straightforward in this case and consists of normalizing the sample counts obtained from the expectation step such that they form a probability distribution across the symbols of the alphabet for each column of the PSSM.

The most well-known EM-based motif discovery still in use today is MEME [5]. MEME makes use of a number of extensions to the basic EM-based algorithm, including relaxation of the assumption of one motif per sequence to adapt to noisier data with zero or more than one copies of a motif per sequence, also motifs are probabilistically erased after they are found to allow for several distinct motifs to be found within the same set of sequences.

3.3.3 Existing Gibbs Sampling PSSM methods

Lawrence et. al. [32] were the first to publish a paper related to Gibbs sampling and motif discovery in biological sequences. As with EM-based approaches, the motif is described as

a position specific scoring matrix. In the basic form of the Gibbs sampling approach, one motif per sequence is assumed, a motif width, w, is input along with the set of sequences believed to share a motif. At any stage of the Gibbs sampling procedure, there will be a list of putative start points of the motif, with one motif per sequence. The regions covered by these motifs are aligned (an ungapped alignment of sequences of length equal to w) and a PSSM is obtained by counting occurrences of each symbol in each of the columns independently. The distribution of symbols in all of the regions of all sequences not covered by motifs is also obtained and provides the background distribution of symbols.

The algorithm is initialized by choosing random start sites within the various positions. Two steps are carried out:

a) Predictive Update step One of the input sequences is chosen, either randomly or in a specified order. The PSSM and background frequencies are then calculated based on the selection of motifs in all sequences but the single input sequence chosen.

b) Sampling step In the sequence selected in the predicted update step, every segment of width w is considered as a possible instance of the motif for that sequence. The probability that the sequence was sampled from the background distribution, P(X), is determined, as well as the probability that the sequence was generated by the motif, as encoded by the PSSM from the previous predictive update step, Q(X). A weight A(X) = Q(X) / P(X) is then assigned to each segment, and based on these weights, a segment is sampled. The start point of this segment is then assigned as the start point for motif within that sequence.

Though the initial random configuration of motif start sites should favour no particular motif, once several of the correct start sites of motifs have been randomly chosen, the PSSM generated will imperfectly represent the true description of the motif, and correct start sites in other sequences will be sampled with greater preference.

A number of extensions have been made to the basic Gibbs sampling procedure over the years. Neuwald et. al., [40] relaxed the one-motif per sequence assumption by introducing the concept of motif sampling. Thijs et. al. [53, 54] extended the Gibbs sampler in several ways for DNA sequences only, most notably by making use of a higher order background model. Two extensions involving allowances for gapped probabilistic motifs will be discussed now.

3.3.4 Gibbs sampling: Statistical Perspective

An accessible review of Gibbs sampling is provide in [10]. Gibbs sampling is used in practice when we wish to find characteristics of a marginal densitygiven a joint density $f(x, y_1, ..., y_p)$:

$$f(x) = \int ... \int f(x, y_1, ...y_p) dy_1, ...dy_p. \tag{3.2}$$

In practice, the integral may be analytically difficult or impossible to obtain, so instead a sample from $f(x)$ is taken. In an example with a pair of random variables (X, Y), the Gibbs sampler generates a Gibbs sequence of random variables

$$Y'_0, X'_0, Y'_1, X'_1, Y'_2, X'_2, ...Y'_k, X'_k. \tag{3.3}$$

An initial value Y'_0 is specified, and conditional distributions $f(x|y)$ and $f(y|x)$ are alternatively sampled:

$$\begin{aligned} X'_j &\sim f(x \mid Y'_j = y'_j) \\ Y'_{j+1} &\sim f(y \mid X'_j = x'_j). \end{aligned} \tag{3.4}$$

The distribution of X'_k converges to $f(x)$ as $k \leftarrow \infty$.

In the motif discovery with PSSM case f(x) can be thought of as the potential distribution of motif start points (with fixed length PSSM motifs) within one of the sequences (the "leave-out" sequence). The variables in the set $y_1...y_p$ are the motif start points in the remaining sequences. We determine $f(x|y_1...y_p)$ by obtaining the PSSM from the motifs in the remaining sequences and the background vector from the non-motif regions, from which we can readily obtain the $Q(X)/P(X)$ scores of motif / background probability for each candidate motif in the leave out sequence; once this is normalized we will have the density $f(x|y)$ from which we may sample. In general, as we are working with more than two sequences, and we are interested in the motifs present in not only one but all of the input sequences, each sequence will be alternately taking a turn as the "leave-out" sequence, or the sequence from which we sample a motif from the conditional distribution $f(x|y)$.

While we could obtain such values as the frequency with which each potential start site was sampled, Gibbs sampling in motif discovery is mainly used as a tool in which a motif site is sampled for each sequence, and the combination of start sites giving a high score or likelihood is chosen as the best overall motif representation.

3.3.5 Gibbs sampling vs. Expectation Maximization

Depending on one's perspective, EM may be thought of as the deterministic analog of Gibbs sampling, or Gibbs sampling may be thought of as the stochastic, non-deterministic analog of EM. Gibbs sampling is less prone than EM to local optima because it may at certain stages sample a less than optimal choice of parameter. However, convergence is more difficult to determine in a Gibbs sampling approach [47].

3.3.6 MCMC Method

Neuwald and Liu [39] make use of a Markov Chain Monte Carlo method for exploring the search space of possible alignments. A full alignment (multiple copies of different motifs) is carried out, conserved, ungapped block regions initially being separated by variable length regions not involved in the alignment. The approach maintains an ungapped alignment throughout the entire process, but individual proteins may be shifted within the alignment, or individual residues within proteins may be inserted or deleted in order to make the alignment a higher scoring one. A traceback needs to be kept so that one can recover the original alignment, complete with gaps. Though this is a multiple alignment procedure, it is also a motif discovery algorithm, as the blocks can be shifted and correspond to the most highly conserved regions.

The alignment probabilistically transitions to new states, which are first proposed and then accepted or rejected according to the probability ratio between the proposed and current state. Given a current alignment, transitions to other states can include adding or deleting columns, hiding an insertion (the opposite being showing the insertion), which involves removing a portion of a sequence from the block alignment that corresponds to a short insertion within a conserved motif. Likewise, the Fill Deletion operation fills in gaps within a sequence in order for it to better correspond to actual evolutionary history.

No implementation of this approach is available, or compared in the recent motif paper by Frith et. al. [18] and it is difficult to understand the practical ramifications of their approach with datasets. Frith et. al, however, point out several shortcomings of their approach. Two notable shortcomings, which are addressed both by Frith et. al and by our own work by virtue of the profile HMM technology include the fact that Neuwald and Liu use simple Dirichlet priors rather than a mixture for amino acid frequencies, and that Neuwald and Liu do not use position specific insertion and deletion probabilities. Since

insertions and deletions tend to concentrate in a few positions, position specific insertion and deletion probabilities are more realistic. As can already be seen from the discussion of profile-HMMs, these issues are addressed by our profile-HMM based approach, as each column can vary in terms of the transitions to insertion and deletion states.

3.3.7 GLAM2

Frith et. al [18] developed GLAM2. Starting from a random alignment corresponding to a motif in the sequence, two types of Gibbs sampling are performed that modify the alignment. In site sampling, one of the input sequences is chosen at random and re-aligned to the motif. Via dynamic programming-based alignment, all possible continuous substrings of the sequence are aligned to the motif, with a score M(i,j) recorded for the score obtained by aligning the subsequence starting at position i and ending at j. The sum of all of these scores is taken, and the choice of sampling from within the protein sequence and not sampling is made by summing all of the endpoint scores, and assigning an arbitrary score of 1 to the option of not sampling. Then a sampling is done in which the choice is between not sampling that sequence at all (score is 1), or sampling from the protein sequence (score is sum of endpoint scores). If the endpoints scores are higher, the chances of excluding the sequence from the model will be lower, as the score of 1 will be lower relative to the sum of scores.

If the decision to sample from the protein sequence for this iteration is made, the endpoint, j is sampled proportional to the sum of all scores ending at j. Finally, an alignment with endpoint j is sampled, using a stochastic traceback. In column sampling, a column in the motif alignment is chosen at random and removed, then each of the other non-motif columns is considered as a possible column within the motif, with sampling done proportional to the alignment score that would result.

Column sampling can increase or decrease the width of the alignment chosen, and is thus used by GLAM2 to optimize the motif width. Non-motif containing, contaminating sequences are, in principle, more likely to be excluded than motif containing sequences by site sampling. If we assume that the correct motif has been sampled from at least most of the motif-containing sequences, when the model of the motif is aligned to a leave-out, non motif-containing sequence, the expected score for any endpoint j will be relatively low, and the arbitrary score of 1 that is assigned to excluding the sequence will be relatively high, and more likely to exclude this sequence compared to a motif-containing sequence where

the sum of scores for endpoints will be relatively high compared to the score of 1 for leaving the sequence out from the current iteration.

GLAM2 makes use of a motif representation system very similar to that seen in profile-HMMs. The most notable difference pointed out in their paper was the fact that insertion probabilities do not distinguish between insertion-opening and insertion-extension (corresponding to the match $\leftarrow$ insert and insert $\leftarrow$ insert transition probabilities of the profile-HMM, respectively).

Due to the very recent publication of GLAM2, it is more difficult to address any shortcomings in their work. Of note is the fact that initially all sequences are included in a random alignment. Sequences may then be probabilistically removed from the alignment (and subsequently added again) in the site sampling stage. This could be called a top-down method of selecting sequences that contribute to the motif, in contrast to a bottom-up method that starts with an alignment of only a subset of sequences and adds others in a probabilistic fashion.

In cases in which all of the input sequences contain a motif that can be detected, the question of selecting only a subset of sequences to be involved in a motif representation is irrelevant. However, if a subset of the input sequences does not share a common motif with the remainder, this becomes a more relevant question. Some of the comparative experiments will address this concern.

3.3.8 How is our work different?

First and foremost this was designed to be an extension of the Gibbs Sampler, an intuitive, elegant, and prolific method for ungapped motif detection, to the case where motifs with gaps present in a subset of the sequences can also be effectively and automatically discovered. The subcellular localization prediction problem served as initial motivation, as known type II secretion motifs are variable in length. In light of the work of Frith et. al., [18] new testing datasets with functions unrelated to localization were made publicly available. When addressing the problem of some input sequences not containing a motif common to the other ones, our method also includes two distinctive methods of sequence selection that are a bottom-up approach in contrast to GLAM2's top-up approach.

Chapter 4

Methods

4.1 Datasets

4.1.1 Localization motifs

Protein subcellular localization is a prime example of the biological process which may involve the type of protein motifs we wish to find. The type II secretion [44] system in bacteria relies on motifs located at the extreme N-terminus (starting at amino acid index 1) of the protein for the crossing of the bacterial inner membrane. Unlike functional motifs, these signal sequences are short and have a wide variability in length, and are therefore a good candidate for this type of motif discovery. Type III [30] and type IV [11] secretion systems in bacteria also make use of their own distinctive machinery, and the proteins exported via these mechanisms do not, as of yet, have a well characterized common motif responsible. Type II secreted proteins can serve as a positive set of proteins with known motifs planted, whereas motif discovery in type III and type IV secreted proteins represents a search for heretofore undiscovered motifs.

A thorough search of published research and review articles on protein secretion was done to identify as many type III and type IV secreted proteins as possible. Type II secreted proteins, on the other hand, are already identified in the UniProt [13] database, based on having a "Signal" domain present in the Sequence Annotation field for the protein's record.

A total of 207 Gram negative type II secreted proteins were recovered, 100 type III secreted proteins, and 41 type IV secreted proteins. Each of the datasets were found to have biases in terms of the proteins, i.e. multiple proteins performing the same function

in very similar strains of bacteria. Due to the redundancy of the UniProt database, many of these proteins would be expected to be extremely closely related proteins from similar strains of bacteria, and share global levels of similarity, rather than the short, local motif level of similarity we are looking for. Therefore, removal of redundant proteins from a dataset prior to motif discovery is an indispensible process. CD-HIT [34] was chosen as the data reduction algorithm to use. This program takes as input a set of proteins and an identity percentage threshold, and clusters the input proteins; proteins are put in the same cluster if their percent identity is above the threshold. The final output file only the largest protein in each cluster. Using an identity threshold of 40% (No two proteins in the dataset more than 40% identity) resulted in sets of size 158, 74, and 27 for types II, III, and IV secretion, respectively.

Because all type II secreted proteins have an extreme N-terminal secretion motif (that is, the motif starts at the amino acid with index 1), any possibility of the motif detection algorithm favoring similar regions with similar indices was eliminated by randomly planting the region annotated in UniProt as Signal anywhere else within the sequence.

In the process of developing an updated version of PSORTb, developers also made available a set of known Gram negative bacterial cytoplasmic proteins. 5025 total proteins were recovered, of which 3544 remained following a reduction with the program CD-HIT removing any proteins with greater than 80% identity to a protein already in the dataset. These proteins were used as negative cases, when needed, in classification compared to other proteins having a secretion motif, as the cytoplasm is presumably the default location of proteins that are synthesized at this location but lack a signal needed for translocation to another compartment.

Type II secreted proteins were also involved in an experiment where proteins with a known, common motif were input to the motif discovery algorithm along with sequences known not to contain the motif; the non-motif containing proteins were randomly selected from the Gram-negative cytoplasmic set. Two ratios were tested, where the cytoplasmic proteins comprised 20%, and 33% of the total set of proteins input to the algorithm.

4.1.2 PROSITE motifs

Frith et. al [18] made use of 58 sets of proteins with a total of 368 sequences with each set containing only proteins that matched a particular PROSITE regular expression. Sequences matching more than once to the same PROSITE expression in different

subregions were excluded from the datasets. Highly similar sequences were also removed from sets to arrive at the final total of 368 sequences using the program BLASTCLUST (ftp://ftp.ncbi.nlm.nih.gov/blast). These sets of proteins were used exactly as Frith et. al. used them, to comparatively assess the alignment accuracy of the resulting best alignment derived from various program.

Similar to the type II study above with non-motif containing proteins being included in the set of input sequences, three of the largest sets of PROSITE pattern containing proteins were further studied in this manner. The pattern identifiers are: PS00028 (zinc finger domain signature, 38 proteins), PS60014 (alpha-conotoxin, 17 proteins), and PS01159 (WW domain, 13 proteins). The non-motif containing proteins that were combined with motif containing sequences in the input dataset were randomly drawn from UniProt.

4.2 Basic Gibbs sampling algorithm (HMM-Gibbs)

The required input to the Gibbs sampling approach is a set of proteins believed to share a common motif, as well as an estimated minimal motif width w. The default minimal width used in experiments was 12 amino acids, as this is below the length of most of the PROSITE motifs and the type II secretion motif, and the usefulness of a profile HMM derived from anything shorter than this is questionable.

The basic outline of the algorithm is presented in algorithm 4.1. Several helper functions are listed following this algorithm 4.2 outlines the leave one out procedure used for scoring a currently sampled model algorithm 4.3 and algorithm 4.4 outline how the background distribution is obtained, and how a background probability of subsequences is obtained.

Several approaches are not listed as separate algorithms. The hmmbuild procedure is that used by HMMER to build profile HMM given a multiple sequence alignment, based upon the counts of amino acids in each column. HMMER's Maximum A Posteriori is used to determine the actual architecture of the model, locations of insert columns vs match columns. The msa subroutine calls, in practical terms, the method PROMALS and returns a multiple alignment of the currently sampled substrings, excluding the leave-one-out.

The steps followed in the most basic procedure are as follows:

a) Sample uniformly a set of random start points for the motifs, one random start point per sequence. The initial motifs will consist of the set of motifs that start at these random start points and are of width w.

b) One sequence is designated as the leave-out sequence. The currently sampled motifs from all sequences but the leave-out sequences are multiply aligned using the program PROMALS.
c) A background null model from the all sequences but the leave-out sequence is generated. This is a simple model consisting of proportion of each amino acid present in the sequences, excluding the regions covered by the sampled motif regions.
d) Using the HMMER program, a profile-HMM is built from the multiple alignment of sampled subsequences (excluding the subsequence from the leave out subsequence).
e) From the leave out sequence, all candidate subsequences with a width ranging from $w4$ to $w + 4$ are extracted, and aligned to the profile HMM built in the previous step. Coming up with a suitable range of widths to sample is a fairly ad hoc parameter. The choice of 4 was used as a balance between too low (motifs with larger number of insertions or deletions would be missed) and too high (too large of a search space from which to sample).
f) For each candidate subsequence, the score Q(X), the probability that the subsequence was generated by the profile HMM is calculated, using either the Forward algorithm, or the Viterbi approximation.
g) For each candidate subsequence, the score P(X), the probability that the subsequence was generated by the background null model from step c is calculated.
h) The ratio Q(X) / P(X) is determine for each candidate subsequence. These ratios are then normalized to sum to 1, thus forming a probability distribution.
i) Based on the normalized ratios in step h), a subsequence is sampled and this is considered the current motif for the leave out sequence. If there still remains a sequence that has not yet been sampled for the current mid-level iteration, return to step b) using a different sequence as the leave-out sequence. If all sequences have had a motif sampled exactly once for the current round, proceed to step j).
j) If less than 50 iterations of steps b) through i) have been carried out for the given start set, return to step b).
k) After 50 iterations of steps b) through i), return to step a), select a different random set of start points, increase motif width w by 2.

The procedure involves several nested levels of iteration. One round of sampling for one particular sequence (steps b) through j)) is the innermost iteration. Carrying out steps b) through) once for each of the input sequences is the middle level of iteration. In practice, the algorithm is run with 10 different random start configurations (and motif widths), one

run of steps a through i (with steps b through i repeated multiple times) is the outer level of iteration.

Model selection (below) is carried out after all iterations are complete, to select the best model, according to a leave one out procedure. Several forms of output are now available and which one is used may vary depending on the data used.

a) The profile-HMM itself. HMMER has implementations in place that allow this to happen. Given a protein sequence, it is a matter of scoring the sequence against the model using the Viterbi or Forward algorithm, sequences that score above a certain scoring threshold are candidates for further study as to whether they truly contain the motif.

b) A multiple sequence alignment. Profile-HMM's are based upon multiple sequence alignments. Though this sampling algorithm makes use of profile-HMM's because of their expressivity when it comes to insertions and deletions, another form of motif representation may be built upon the multiple sequence alignment.

c) The start and end points of the motifs within the sequences themselves. Rather than relying on the model or the alignment, the user may simple be interested in having a general sense of the motif locations and the general amino acid composition of the areas in question; this information is still useful for further biological validation.

4.2.1 Model selection

Refer to algorithm 4.2.

At every mid-level of iteration (a subsequence for each input sequence is sampled exactly once) a profile HMM motif model can be generated from a multiple alignment of the currently selected subregions of all of the input sequences. During the algorithm run, a selection of subsequences corresponding to a true biological sequence may be sampled, but due to the sampling procedure, in one or more sequences, an incorrect subregion is sampled in a subsequent iteration, thus leading to a decrease in the quality of the model that is built. Because there is no guarantee of improvement in the model from one iteration to the next, criteria for selecting the best possible model from all iterations are needed.

The approach chosen was, at the end of each mid-level iteration, to do a leave-one-out evaluation of the currently selected model. Each sequence is, in turn, treated out as a leave out sequence, and a profile HMM is built from the alignment of the remaining motif subsequences. Similar to the actual Gibbs sampling stage, the scores Q(X) and P(X) are determined, and the ratio is taken. After all iterations are complete, the model with

the highest average ratio per sequence is chosen as the final output model, along with its corresponding multiple sequence alignment.

Algorithm 4.1 HMM-Gibbs Basic motif discovery

Input: set of Proteins S, minimal motif width w

Output: top scoring model *maxmodel*, corresponding multiple alignment *maxmsa*, motif start points for each protein *AlignStarts*, and motif end points *AlignEnds*

1: $maxScore \Leftarrow -\infty$, $cw \Leftarrow w$
2: **while** $cw \leq w + 20$ **do**
3: **for all** $p \in S$ **do**
4: $AlignStarts_p \Leftarrow rand(1..Length(p) - cw)$
5: $AlignEnds_p \Leftarrow AlignStart_p + cw - 1$
6: **for all** $iter \in 1..50$ **do**
7: **for all** $p \in S$ **do**
8: $T \Leftarrow S - p$
9: $b \Leftarrow background(T, AlignStarts, AlignEnds)$
10: $msa \Leftarrow multiplyAlign(T, AlignStarts, AlignEnds)$
11: //HMMER's hmmbuild takes as input a multiple sequence alignment outputs
12: // profile-HMM.
13: $h \Leftarrow hmmbuild(msa)$
14: // Viterbi, as discussed in Chapter 3 returns the probability of the
15: // most probable state path through an hmm
16: **for all** substring ss of p such that $cw - 4 \leq Length(ss) \leq cw + 4$ **do**
17: $Viterbi_{ss} \Leftarrow Viterbi(ss, h)$
18: $Background_{ss} \Leftarrow BackgroundProbability(ss, b)$
19: $OverallScore_{ss} \Leftarrow Viterbi_{ss}/Background_{ss}$
20: $New\ motif\ m \Leftarrow Sampled\ proportional\ to\ OverallScore$
21: $AlignStarts_p \Leftarrow$ index of first residue of m in p
22: $AlignEnds_p \Leftarrow$index of last residue of m in p
23: $score_{iter} \Leftarrow LeaveoneOut(S, AlignStarts, AlignEnds)$
24: **if** $score_{iter} \geq maxscore$ **then**
25: $maxscore \Leftarrow score_{iter}$
26: $maxmsa \Leftarrow multiplyAlign(S, AlignStarts, AlignEnds)$
27: $maxmodel \Leftarrow hmmbuild(maxmsa)$
28: $cw \Leftarrow cw + 2$
29: **return** $maxmsa, maxmodel, AlignStarts, AlignEnds$

Algorithm 4.2 Leave-one-out scoring of an alignment, used for model selection

Input: set of Proteins S, corresponding motif start points *AlignStarts*, motif end points *AlignEnds*

Output: *score* resulting from leave-one-out analysis of motifs

1: $score \Leftarrow 0$
2: **for all** $p \in S$ **do**
3: $T \Leftarrow S - p$
4: $b \Leftarrow background(T, AlignStarts, AlignEnds)$
5: $msa \Leftarrow multiplyAlign(T, AlignStarts, AlignEnds)$
6: $h \Leftarrow hmmbuild(msa)$
7: $ss \Leftarrow p[AlignStarts_p..AlignEnds_p]$
8: $Viterbi_{ss} \Leftarrow Viterbi(ss, h)$
9: $Background_{ss} \Leftarrow BackgroundProbability(ss, b)$
10: $OverallScore_{ss} \Leftarrow Viterbi_{ss}/Background_{ss}$
11: $score \Leftarrow score + OverallScore_{ss}$
12: **return** *score*

Algorithm 4.3 Background

Input: Set of full protein sequences S, vectors of where motifs start and end *alignStarts* and *alignEnds*

Output: Vector b of 20 amino acid probabilities

1: **for all** $a \in 20$ *letter amino acid alphabet* **do**
2: $b[a] \Leftarrow 0$
3: $totalCount \Leftarrow 0$
4: **for all** $PinS$ **do**
5: $i \Leftarrow 0$
6: **while** $i \leq P.length()$ **do**
7: **if** $i < alignStarts_P$ *or* $i > alignEnds_p$ **then**
8: $b[P[i]] \Leftarrow b[P[i]] + 1$
9: $totalCount \Leftarrow totalCount + 1$
10: **for all** $a \in 20$ *letter amino acid alphabet* **do**
11: $b[a] \Leftarrow b[a]/totalCount$
12: **return** v

Algorithm 4.4 BackgroundProbability

Input: Protein String s, 20 element, 1 dimensional background model b

Output: Probability p of b generating s

1: $i \Leftarrow 1$

2: $p \Leftarrow 1$

3: **while** $i \leq s.length()$ **do**

4: $p = p * b[s[i]]$

5: **return** p

4.2.2 Implementation details

PROMALS [42] was selected as the multiple alignment algorithm to be used in step b) based upon its easy availability and its state-of-the-art usage of predicted secondary structures and PSI-BLAST [3] derived profiles to guide the alignment, compared to other multiple sequence alignment algorithms such as ClustalW [55] and MUSCLE [17]. Poor results were noted in practice with the latter two multiple sequence alignment steps with either the type II motifs or PROSITE patterns.

For a full description of the PROMALS multiple sequence alignment method, see [42]. In short, sequences are taken as input (in the case of our approach, short subsequences), highly similar sequences are progressively aligned using a fast way, for other sequences, a PSI-BLAST search with three iterations and an E-value cutoff of 0.001 is done against the UniRef database. This is a form of search that finds distantly related proteins and returns a position specific scoring matrix based upon the protein. The PSI-BLAST checkpoint file is used to predict secondary structure using the PSIPRED method [29]. Profiles are obtained from the PSI-BLAST PSSM and the predicted secondary structures obtained, and the sequences are aligned in a progressive fashion, based on these profiles. In our implementation, the short substrings being input to the program would be unlikely to generate realistic PSI-BLAST based PSSM's, or secondary structure prediction, so this information is obtained from whole protein sequences before any multiple alignment takes place, and the alignment takes place from only the regions of the PSSM and secondary structure prediction that correspond to the substring currently sampled for a protein.

A global-to-global alignment was done between candidate subsequences and the generated model in step d). This avoids cases of local alignment where the first half of the

candidate subsequence is aligned to the second half of the profile-HMM (or vice versa). This results from the alignment looping through states N or C of the profile HMM of Figure 3.1 with very little reduction in the final probability, leading to artificially high Q(X) values for cases when a local alignment is done.

4.3 Extensions of the basic Gibbs sampling procedure

4.3.1 Subset selection for motifs to sample

An observation that came after working with synthetic datasets was in cases where 50 sequences with planted motifs were input into the Gibbs sampling procedure, and less than 10 sequences had the correct planted motif sampled, perhaps by chance, this was not enough to influence the parameters of the profile-HMM being generated to actually lead to other motifs being sampled correctly. This becomes even more of a concern in which some of the proteins submitted to the procedure do not contain any of the motifs in question, and no possible subsequence sampled from the sequence can bring the motif model to a closer approximation of the true motif. This serves as motivation for finding a way of building a profile-HMM from only a subset of the selected sequences, and finding a good way of choosing such a subset.

If we have a set of X sequences submitted to the Gibbs sampling procedure, and only Y $<$ X of the sequences have the correct motif being sampled, those Y true motifs would be expected to have higher level of similarity to each other compared to subsequences drawn at random from other sequences. We therefore establish a criteria for selecting a subset of subsequences with greater similarity to each other than the remainder.

At each stage of the Gibbs sampling procedure, before the profile HMM is built from a set of subsequences, the background model is created from the non-motif region of all sequences and a profile HMM is built individually from each of the currently sampled subsequences (except for the currently left out sequence).

As an aside to this method, it should be kept in mind, that, while profile-HMMs typically represent a multiple alignment of sequences, they can easily represent a single sequence, against which other sequences may be aligned. Though the actual sequence contains no gaps and 100% probability for a single amino acid in each of its columns, the Dirichlet priors for emission probabilities will generate pseudocounts for amino acids in each position of the column, with higher pseudocounts for amino acids with higher biological substitution

probability than the observed amino acid, corresponding to the amino acid substitution matrices used commonly in pairwise sequence alignment algorithms [25]. Non-zero transition probabilities to insertion and deletion states will also be present in each column, due to transition priors for insertion and deletion states, these will be of a lower probability compared to direct transition from match to match (ungapped alignment) and therefore correspond to the gap penalties used in pairwise alignment. It was better to make use of existing profile-HMM implementation for pairwise alignment then attempting to write or modify a traditional Smith-Waterman based approach for pairwise alignment when the two are really corresponding problems.

Algorithm 4.5 HMM-Gibbs Subset selection version

Input: set of Proteins S,

Output: top scoring model maxmodel, corresponding multiple alignment maxmsa, motif start points for each protein AlignStarts, and motif end points AlignEnds

1: $maxScore \Leftarrow -\infty$
2: $cw \Leftarrow w$
3: **while** $cw \leq w + 20$ **do**
4: **for all** $p \in S$ **do**
5: $AlignStarts_p \Leftarrow rand(1..(Length(p) - cw))$ //random int from 1 to length - cw
6: $AlignEnds_p \Leftarrow AlignStart_p + cw - 1$
7: **for all** $iter \in 1..50$ **do**
8: **for all** $p \in S$ **do**
9: $T \Leftarrow S - p$
10: $b \Leftarrow background(T, AlignStarts, AlignEnds)$
11: $proteinSelection = subsetSelect(T, AlignStarts, AlignEnds)$
12: $msa \Leftarrow multiplyAlign(proteinSelection, AlignStarts, AlignEnds)$
13: $h \Leftarrow hmmbuild(msa)$
14: **for all** $substring\ ssofpcw - 4 \leq Length(ss) \leq cw + 4$ **do**
15: $Viterbi_{ss} \Leftarrow Viterbi(ss, h)$
16: $Background_{ss} \Leftarrow BackgroundProbability(ss, b)$
17: $OverallScore_{ss} \Leftarrow Viterbi_{ss}/Background_{ss}$
18: $New\ motif\ m \Leftarrow Sampled\ proportional\ to\ OverallScore$
19: $AlignStart_P \Leftarrow\ index\ of\ first\ residue\ of\ m\ in\ p$
20: $AlignEnd_p \Leftarrow indexoflastresidueofminp$
21: $score_{iter} \Leftarrow Leave - one - out(S, AlignStarts, AlignEnds)$
22: **if** $score_{iter} \geq maxscore$ **then**
23: $maxscore \Leftarrow score_{iter}$
24: $maxmsa \Leftarrow multiplyAlign(S, AlignStarts, AlignEnds)$
25: $maxmodel \Leftarrow hmmbuild(maxmsa)$
26: $cw \Leftarrow cw + 2$

Algorithm 4.6 subsetSelect

Input: set of Proteins S, corresponding motif start and end points $AlignStarts$, $AlignEnds$

Output: $U \subseteq S$, proteins whose substrings form a connected component

1: //ScoreMatrix tracks pairwise scores between pairs of proteins, initialized to $-\infty$
2: $b \Leftarrow background(S, AlignStarts, AlignEnds)$ // Obtain background model
3: **for all** $p \in S$ **do**
4: $ss \Leftarrow p[AlignStarts_p..AlignEnds_p]$ //Substring of protein p corresponding to motif
5: $T \Leftarrow S - p$
6: $h \Leftarrow hmmbuild(ss)$ //Building a model not from an alignment, but a single protein
7: **for all** $q \in T$ **do**
8: $ssq \Leftarrow q[AlignStarts_q..AlignEnds_q]$ //Substring of q corresponding to motif
9: $Viterbi_{ssq} \Leftarrow Viterbi(ssq, h)$
10: $Background_{ssq} \Leftarrow BackgroundProbability(ssq, b)$
11: $OverallScore_{ssq} \Leftarrow log(Viterbi_{ssq}) - log(Background_{ssq})$
12: **if** $ScoreMatrix[p][q] < OverallScore$ **then**
13: $ScoreMatrix[p][q] \Leftarrow OverallScore$, $ScoreMatrix[q][p] \Leftarrow OverallScore$
14: $scoreThreshold \Leftarrow 10$
15: **while** $scoreThreshold \geq -5$ **do**
16: $g \Leftarrow BuildGraphBasedOnScores(ScoreMatrix, scoreThreshold)$
17: **if** g *contains connected component with* $|v| \geq vertexThreshold$ **then**
18: $cc \Leftarrow$ *largest connected component in* g, $R \Leftarrow$ *set of proteins with a vertex in* cc
19: $W \Leftarrow S - R$
20: $msa \Leftarrow multiplyAlign(W, AlignStarts, AlignEnds)$, $h \Leftarrow hmmbuild(msa)$
21: **for all** $P \in W$ **do**
22: $ss \Leftarrow p[AlignStarts_p..AlignEnds_p]$
23: $Viterbi_{ss} \Leftarrow Viterbi(ss, h)$, $Background_{ss} \Leftarrow BackgroundProbability(ss, b)$
24: $OverallScore_{ss} \Leftarrow log(Viterbi_{ss}) - log(Background_{ss})$
25: **if** $OverallScore_{ss} \geq scoreThreshold$ **then**
26: $R \Leftarrow R \cup \{P\}$
27: **return** R
28: $scoreThreshold \Leftarrow scoreThreshold - 1$
29: **return** S //No high scoring component, return full set by default

Algorithm 4.7 BuildGraphBasedOnScores

Input: scoreMatrix, scoreThreshold

Output: graph G

```
1: for all P ∈ scoreMatrix.rows do
2:   G.addVector(P)
3: for all P ∈ scoreMatrix.rows do
4:   for all Q ∈ scoreMatrix.columns do
5:     if scoreMatrix[P][Q] ≥ scoreThreshold then
6:       G.addEdge(P, Q)
7: return G
```

Returning to the method, the protein is run using PSI-BLAST (3 iterations with an E-value of 0.001) and the full aligned set of hits in the motif region under consideration is used instead of only the observed protein subsequence to build the profile-HMM. Every sampled subsequence is then run using the Viterbi algorithm against every profile HMM generated from a corresponding single subsequence. This generates an approximation of the probability that the single subsequence profile HMM generated each subsequence. Every sample subsequence is then run against the background model to determine the probability of being generated by the background model, and the probabilities divided: Viterbi probability divided by background probability, and the logarithm obtained. What we essentially have for every pair of subsequences s and t is a score for the subsequence s aligning to the model from subsequence t vs the background model. We will also have a score for the subsequence t aligning to the model from subsequence s vs the background model. This is not guaranteed to be a reflexive relationship, that is these two probabilities are not expected to be equal, in which case we choose the higher score of the two to represent the score associated with this subsequence pair.

This problem can be then thought of as a graph, with the sampled subsequences being nodes, and edges between them being present if the score obtained upon their alignment using the profile-HMM based technique is above a certain threshold. It is not necessarily the best to use a fixed threshold for the score at every iteration, as during earlier iterations when essentially random subsequences are being sampled, the similarity between sampled subsequences would be very low, but would be expected to increase at later iterations if the Gibbs sampling procedure is working properly.

Therefore, at every iteration, scores are obtained for all pairs of subsequences. Graphs will be built using each subsequence as a vertex, with an edge between the two vertices if the score is greater than a threshold. A strict score threshold is initially used, and a check is done if a connected component with vertices greater than a constant threshold number of vertices (total number of proteins / 10, or 3, whichever is greater) is present in the graph. If not, the score threshold is reduced, edges meeting the new lower threshold criteria are added, and another check is done for connected components meeting the threshold number of vertices.

The vertex threshold is chosen somewhat arbitrarily; when we have an imperfect motif model in early stages it is probably more appropriate, as only a minority (eg. 1/10) of the proteins may have a correctly sampled motif, at later stages it may be more appropriate to raise the vertex threshold, thus requiring a larger graph. On the other hand, it may in other cases be appropriate to keep the vertex threshold low, when there will be a smaller, but higher scoring graph present. Further experimentation and optimization regarding the vertex and score threshold, and how they interact, are needed.

In an effort to add other subsequences that were missed by the pairwise approach but have a high score to the emerging model, the model with the subset selected is built, and aligned against all remaining substrings. Those that score above the threshold needed to obtain the connected component are added to the model.

Once the subsequences of the connected component exceeding the threshold number of vertices have been obtained, all possible subsequences from the left-out sequence are scored against the motif model and the background model, but the motif model will only involve subsequences from proteins with a corresponding vertex in the connected component.

The full outline of the modified Gibbs sampling approach is listed in algorithm 4.5 with the procedure subsetSelect outlined in algorithm 4.6 and an auxiliary function for forming the graph provided in algorithm 4.7.

4.3.2 Improved initial start point selection using subsequence clustering

Based on the assumption that in certain cases conserved motifs of interest may share certain physicochemical properties that distinguish them from surrounding areas, means of using non-random start point selections at the beginning of each outer-level iteration were developed. The Amino Acid Indices [31] are sets of real valued properties that have been experimentally determined for each of the amino acids. Though higher order properties

emerge from a protein when amino acids are strung together and folded in a three dimensional conformation that go beyond the average of these properties, they can still be a useful approximation of what is happening locally in sequences, and have been previously used as a classifier for protein localization [9]. When averaged across the set of amino acids, motifs, in certain cases such as the type II signal are different in terms of the properties compared to other regions of the motif. Based on the average of physico-chemical properties of amino acids within them, subregions of the proteins are clustered, and certain clusters are selected as sets of motifs used to initialize the Gibbs sampling procedure.

Every possible subregion of length w cannot be simply input into a clustering algorithm such as k-means; if one particular subregion w1 of length w is shifted over by w to form subregion w2, the two regions will share w-1 amino acids, and the averages of physico-chemical properties taken across their constituent amino acids will be very similar. Thus, the sliding windows do not overlap. In a particular sequence, a different starting point for the sliding window (from 1 to w) will lead to a different set of subsequences being chosen. Unfortunately, due to the great variation in protein length and content, the closest possible subregion of another sequence may result from any of the w possible window starting point selections, leading in total to w^n possible window choices across all of the sequences. For each sequence, a random selection of start point between 1 and w was made, and the clustering approach was repeated with different window start point selections. In practice a total of 10 runs of algorithm 4.8 are done, which in turn provide 10 sets of start points. The end points are a fixed distance from the start point. Proteins that did not have a representative subsequence in a particular cluster would not be used for that particular iteration. Each set of start points replaces the random selection of start points used in algorithm 4.1.

Algorithm 4.8 Clustering Based Subset Selection

Input: *set of proteins S*,
amino acid indices *aaindex* with every aaindex[k] being a vector of 20 real values of chemical properties of amino acids
motif width w

Output: *set of proteins R*,
vector of start indices *alignStarts*,
vector of end indices *alignEnds*

```
1: //sv will be a set of vectors that are onput to k-means
2: //with each vector corresponding to a protein substring
3: //note that all substrings of all proteins are put into
4: //one run to k-means
5: setofVectorssv
6: for all P ∈ S do
7:   substringIndex ⇐ rand(1..w)
8:   while substringIndex < P.length − w do
9:     ss ⇐ P[substringIndex..substringIndex + w]
10:    // v will be the vector of average chemical values for the substring
11:    vectorv
12:    for all k ∈ 1..aaindex.length do
13:      for all aminoacidaa ∈ ss do
14:        v[k] ⇐ v[k] + aaindex[k][aa]
15:      v[k] ⇐ v[k]/w
16:    sv ⇐ sv ∪ v
17:    substringIndex ⇐ substringIndex + w
18: //Now the substrings of all proteins will be clustered
19: //based on average physicochemical properties of residues
20: setofclusterssoc ⇐ kmeans(sv)
21: clustertop ⇐ topCluster(soc)
22: //For every protein with exactly one substring in top
23: //Add the protein to R, and the corresponding start and end
24: //Of the substring to alignStarts and alignEnds
25: //For proteins with more than one substring in top
26: //Randomly select one and proceed
27: return R
```

Algorithm 4.9 topCluster

Input: a set of clusters *soc* with each cluster being a set of substrings

Output: top scoring cluster *top* based on leave one out analysis

1: $maxScore \Leftarrow -\infty$
2: **for all** $c \in soc$ **do**
3: $scores \Leftarrow$ leave-one-out as done in algorithm 4.2
4: **if** $s > maxScore$ **then**
5: $maxScore \Leftarrow s$
6: $topCluster \Leftarrow c$
7: **return** *topCluster*

a) Divide the full set of input sequences into regions having the width of the input motif width parameter (w).

b) Each candidate motif is scored against a set ($\approx$ 500) of real-valued amino-acid physicochemical properties. The properties are for individual amino acids, thus since we are dealing with short peptide regions, the properties are each averaged across the entire motif. Thus each candidate motif is encoded as a vector of 500 real valued properties.

c) The subsequences, from all sequences, are used as input for k-means. K (number of clusters) is chosen such that the average cluster size will be equal to the number of input sequences (one motif per sequence on average).

d) For each selection of starting windows, 10 runs of k-means (with different initial cluster centres chosen) are done.

e) Thus we have a total of 100 different runs of k-means, each one outputting multiple clusters. Each cluster is a candidate for a set of motif start points to run the algorithm. To determine likely start points to move forward with, the score of the start point set is determined in a very similar fashion to what is done at the end of every iteration

f) The top ten clusters are chosen, with the starting points used as the initial motifs as input for the descriptive Gibbs Sampling. (10 different outer iterations of the Gibbs sampling procedure)

g) In cases in which a sequence has more than one representative subsequences in a cluster chosen, one of the representative subsequences is randomly chosen. Sequences having no representative sequences in that cluster were not considered further for that particular outer iteration.

4.4 Types of assessment

4.4.1 Alignment Accuracy

The 58 PROSITE-pattern containing datasets used by Frith et. al [18] are a useful supply of gold standard alignments of short motif regions. From any given motif discovery algorithm, the top scoring (using the method's scoring criteria) motif is chosen. While the representations of motifs output may differ between the algorithms (PSSM, regular expression, or profile HMM), they all represent an underlying multiple sequence alignment of the motif region, and this is what is actually quantitatively assessed. The gapped motif discovery methods GLAM2 and PRATT [28] were used, as well as the basic version of the gapped motif discovery method described in this report. In order to assess how much is actually gained by the added complexity of searching for gapped motifs, two ungapped motif discovery methods, MEME [5] and the original ungapped Gibbs Sampler [32] are used, following default parameters on their webservers (http://meme.sdsc.edu/meme/meme.html and http://bayesweb.wadsworth.org/cgi-bin/gibbs.12.pl?data_type=protein respectively).

Once the multiple sequence alignments from the various methods and datasets were obtained, they were assessed using the same two metrics of Frith et. al. Following their nomenclature, the positive predictive value (PPV) is a measure of the correctly aligned residue pairs as a percentage of the total number of aligned residue pairs in the predicted multiple sequence alignment. Sensitivity is the number of correctly aligned residue pairs as a percentage of the total number of aligned residue pairs in the gold standard alignment.

4.4.2 Detection of Type II Secretion Motifs

There does not exist a gold standard multiple alignment of type II secretion motifs in bacteria. Multiple alignment of motif sub-regions is used as a tool throughout the Gibbs sampling procedure but it is rather as an intermediate step towards building the profile-HMM motif representation from which the P(X) scores come that will guide the next sampling iteration. Therefore, using a PPV or sensitivity score for alignments resulting from motif detection in type II secreted motifs would be impossible. We do, however, know the true cutoff points of the motifs (and the length, since they start at index 1), as they are cleaved off during

the secretion process (though there may exist subtle motifs in the non-signal region of the mature protein as well, these subtle regions, if they lie immediately after the signal region would be separated from the signal region in our datasets as the signal region is moved to a random location within the protein).

From a run of the motif discovery algorithm, rather than the full multiple alignment, the start and end points of each sequence motif contributing to the best scoring motif are recovered. For each sequence, the number of true positive (TP) residues would be the number of signal motif amino acids covered by the region indicated, false positive (FP) residues correspond to non-signal motif amino acids covered by the predicted motif region, and false negative (FN) residues correspond to true signal regions that were not covered by the predicted motif regions. The standard measures of precision and recall are used:

$$Precision = TP/(TP + FP) \tag{4.1}$$

$$Recall = TP/(TP + FN) \tag{4.2}$$

.

If global counts were taken, the numbers would be biased in favour of proteins with longer true motif regions, therefore precision and recall are obtained for each protein individually and averaged across all proteins.

4.4.3 Classification ability of the model

As noted, one of the outputs from the motif discovery algorithm is a profile-HMM. This HMM can be scanned against large databases of proteins, and proteins scoring above a certain score threshold (or below an e-value threshold) are at least candidates for having a motif of comparable biological function. The applicability of motifs discovered using our processes was further assessed, treating the motif extracted as a classifier.

Five fold cross validation was carried out, which involved, in the case of type II secretion motifs, randomly splitting the set of proteins known to carry the motif into 5 equal subsets. 4/5 were used as training data, or input data for the motif discovery algorithm and the best model was chosen. The remaining 1/5 of proteins not involved in training the model were then used as a testing set. They were scored against the HMMER profile-HMM

using HMMERs built-in scoring system. In addition to testing with the 1/5 of true motif-containing proteins, the set of non-motif containing cytoplasmic proteins (reduced with CD-HIT using an 80% threshold) was included as a test set for all 5 folds of the cross validation experiment. This is repeated using each 1/5 of the data as the test data in turn. Various e-value thresholds were used and proteins falling below the threshold were counted as a positive hit for having the motif in question. True positives, false positives, and false negatives were counted at the whole protein level (not the amino acid level as was done with assessment type 2) above), and precision and recall computed. As a comparison, Frith et. al [18] developed GLAM2SCAN, a companion to GLAM2 which takes as input a database of sequences and a GLAM2 output file (which would include all of the relevant motif parameters in their representation), and assigns a score to proteins in the database. A similar training/testing assessment was carried out with GLAM2 replacing our profile-HMM sampling procedure and GLAM2SCAN replacing the profile-HMM scoring procedure.

4.4.4 Testing with non-positive sequences in the training data

As indicated in the introduction to the datasets, from the Type II secretion motifs, PS00028, PS60014, and PS01159 datasets, non-motif containing proteins were added as contaminants to the training datasets, in proportions of either 1/5 or 1/3. Assessments 1, 2, and 3 were carried out with each level of contamination. In the case of assessments 1 and 2, the proteins were added to the full set of proteins, in the case of assessment 3, the proteins were added to the 4/5 training data (for each of the 5 folds of cross validation). The basic implementation of our algorithm assumes that each input sequence contains exactly one input sequence, so assessment 1 would not work favourably with it. However the two add-ons should, if working properly, remove the sequences not containing a motif from the consideration and assessment 1 can be compared with these extensions.

Chapter 5

Results

5.1 Alignment accuracy

As indicated in Table 5.1, there is a clear advantage to be gained when using either our Gibbs-HMM method, or the GLAM2 method, both in terms of sensitivity, (percentage of the gold standard alignment pairs that were predicted by the algorithm) and positive predictive value (PPV, percentage of the alignment pairs predicted by the algorithm that are also alignment pairs in the gold standard alignment). The ungapped method, MEME achieved higher overall PPV than our method, but the sensitivity achieved was very low, indicating that this is a very conservative method that can detect short ungapped true motif regions, perhaps in only a subset of sequences, but missing many predictions in gapped cases.

Looking at individual alignments, it became clear that our Gibbs-HMM method generally aligned to the correct regions, but chose a sub-optimal model length, thus leading to similar PPV but lower sensitivity than seen with GLAM2.

Turning our attention (Table 5.2) now to three of the datasets that were chosen for a

Method	Mean PPV	Mean Sensitivity
Gibbs-HMM	0.574	0.624
GLAM2	0.567	0.752
PRATT	0.326	0.398
MEME	0.620	0.241
Gibbs-Basic	0.4888	0.461

Table 5.1: Comparison of 5 methods' alignment performance with 58 Prosite Datasets

"contamination study" where either 20% or 33% of proteins INPUT to the dataset were randomly selected non-pattern containing proteins, GLAM2 did a very good job of maintaining alignment accuracy and excluding the non-motif containing proteins from the final alignment. The basic Gibbs-HMM method, which does not exclude any sequences from the final alignment, obviously took a penalty in the PPV as many non-true aligned pairs were included. Proteins containing the true motif were also more likely to have a spurious selection of motif.

The subset selection method did help to alleviate the problem, in two of the three cases it even achieved levels of PPV and sensitivity seen with GLAM2, but with one set, PS60014, many incorrect proteins were sampled. It seems with this type of pattern the subsequences were not sampled enough to reach the threshold for the minimum size of a connected component, a lower threshold was picked, and this included many incorrect motifs.

The clustering based selection of starting points (which also selects a subset of the proteins) was attempted with these PROSITE datasets but gave very difficult to interpret results. For examples, with the PS00028 dataset, either "pure" or containing "contaminants", the top cluster with one random run of k-means included substrings from 7 corresponding proteins. Subsequent Gibbs sampling changed the motif location for most proteins very little from the starting points, and inputting the same 7 proteins to GLAM2 gave similar motif locations, rather than an alignment corresponding to the "true" zinc finger motif common to all proteins in the full set. There are clearly some motif-like properties of these regions, but perhaps this was more an artifact of larger functional domains from homologous proteins, though the datasets were reduced to try to eliminate this redundancy. There is no UniProt annotation for the regions covered by these motifs. While there may be something interesting with these proteins, it is not immediately obvious and does not lend itself well to quantitative assessment where we are trying to discover a "true" motif common to the full set of proteins.

The clustering based start point selection is still useful when the true motif, common to all or most proteins, has substantially different physicochemical properties than surrounding areas and is more likely to cluster together between various proteins, as will be seen with type II secretion.

Method	Mean PPV	Mean Sensitivity
Gibbs-HMM 0%	0.551	0.725
GLAM2 0%	0.660	0.761
Gibbs-HMM-Subset 0%	0.625	0.482
Gibbs-HMM 20%	0.451	0.473
GLAM2 20%	0.567	0.752
Gibbs-HMM-Subset 20%	0.465	0.477
Gibbs-HMM 33%	0.392	0.521
GLAM2 33%	0.665	0.772
Gibbs-HMM-Subset 33%	0.428	0.447

Table 5.2: Comparison of 3 methods' alignment performance with 3 of the 58 Prosite Datasets, for each method, there are three types of dataset tested, with 0%, 20% and 33% non motif containing proteins added to the input set

Method	Mean Precision	Mean Recall
Gibbs-HMM	0.891	0.892
GLAM2	0.965	0.918
Gibbs-HMM-Subset	0.903	0.779
Gibbs-HMM-Clustering	0.912	0.707
PRATT	NA	NA
MEME	0.859	0.440
Gibbs-PSSM	0.84	0.675

Table 5.3: Comparison of 7 Methods ability to correctly locate type II Motif in pure input dataset

5.2 Type II secretion motif location

The N-terminal type II secretion motifs were removed from proteins known to have these motifs and planted randomly somewhere within the sequence. Precision/recall studies were then carried out in which these modified proteins were input into 5 motif discovery algorithms and the proportion of residues correctly or incorrectly included in the final output alignment to be part of the motif were counted.

When the dataset consists purely of type II secretion motif-containing proteins, our method falls behind GLAM2 both in terms of precision and recall. This was again the result of differences in terms of length. Upon examining the start site selection in the top scoring model, which in our case had length 24, the predicted motifs did in the vast majority of cases overlap substantially with the true motifs, however, while there was variability in

the length of motifs, it was not enough to cover the full length range of true type II motifs. In cases when the true motif was considerably shorter than 24 amino acids false positive residues were generated from residues wrongly being included in a too-long model, with longer motifs, false negatives were seen from residues being excluded from a too-short model. GLAM2 did a much better job in terms of varying the length of predicted motifs.

When run with the assumption of exactly one motif per sequence (not the webserver default), MEME predicted an ungapped 11 amino acid region in input proteins, which in most cases did overlap with the true motif region.

The PSSM based Gibbs sampler returned a set of results only when choosing the "Motif Sampler" option, as opposed to the simplest "Site Sampler" option. This option instead keeps track of a motif PSSM, and a background model, for each subsequence determines the probability for each of these models, and samples an assignment of the subsequence to one of these models accordingly. A fixed width input is required, and 25 was chosen, as being a reasonable average length of type II secretion motifs.

Due to the stochastic nature, not all runs of the PSSM based Gibbs sampler gave good results, and a high scoring set of results are presented here for their approach. A subset of sequences did have the correct motif region, it seemed that the short 11 amino acid subregion seen with MEME was enough to lead to this region being consistently sampled, but the alignment continued to run in an ungapped fashion (since this is the only form of alignment output by the Gibbs sampler) into regions where the alignment should truly have had a gap.

PRATT struggled greatly to find a descriptive pattern for the type II motifs; the top scoring regular expression was of the form A-x(2)-A indicating two Alanine residues separated by any two amino acids, which generated 512 hits in 126 sequences (multiple matches of this regular expression per protein on average) and other patterns returned were of similar accuracy. Therefore calculations of residue precision or recall for PRATT were not even done.

5.3 Type II secretion motif location with "contaminated datasets"

Moving onto tables 5.4 and 5.5, which compare Gibbs-HMM, the two add-ons, and GLAM2 in their ability to locate true type II Motfs, there are two columns that indicate the precision. In the Mean Precision (Correct Sequences only) column, a residue is counted as

Method	Mean Precision (Correct Sequences only)	Mean Recall	Proportion of Incorrect Sequences Included	Mean Precision (Incorrect Sequences included)
Gibbs-HMM	0.959	0.837	1.00	0.78
GLAM2	0.986	0.880	0.529	0.88
Gibbs-HMM-Subset	0.903	0.829	0.22	0.86
Gibbs-HMM-Clustering	0.912	0.81	0.08	0.89

Table 5.4: Comparison of 4 Methods' ability to correctly locate type II Motif in input dataset with 20% cytoplasmic contaminating proteins

.

Method	Mean Precision (Correct Sequences only)	Mean Recall	Proportion of Incorrect Sequences Included	Mean Precision (Incorrect Sequences included)
Gibbs-HMM	0.925	0.872	1.00	0.69
GLAM2	0.965	0.918	1.00	0.72
Gibbs-HMM-Subset	0.939	0.859	0.36	0.84
Gibbs-HMM-Clustering	0.927	0.877	0.24	0.85

Table 5.5: Comparison of 4 Methods' ability to correctly locate type II Motif in input dataset with 33% cytoplasmic contaminating proteins.

a false positive only if it is within a type II secretion motif containing sequence; this column therefore assesses how well motifs are located within a sequence when contaminating sequences are added. The Mean Precision (Incorrect Sequences Included) column counts as false positive residues any residue predicted to be part of a type II motif in what is actually a non-motif containing sequence, as well as incorrectly predicted residues in motif containing sequences, and is an assessment of the overall precision of the approach. The percentage of cytoplasmic input proteins wrongly included in the final output alignment are listed as proportion of contaminating sequences included. To clarify, the basic Gibbs-HMM method includes exactly one subsequence of EVERY input sequence in the final model, so 100% of the contaminating sequences will be included in the final model. GLAM2, and our two extensions, have means of excluding sequences from the final model, so they are potentially lower than 100%; the lower the percentage the more desirable the result. In our basic Gibbs-HMM approach all input proteins have subsequences listed in the final alignment, so 100% of the "wrong" cytoplasmic sequences are included in the final alignment (as well as 100% of the correct type II proteins). For GLAM2, the numbers were very high, 52.9% when the cytoplasmic proteins formed 20% of the dataset, and 91% while our two extensions achieve comparable precision while including fewer false cytoplasmic proteins into the final alignment. GLAM2 is able to retain high precision in its ability to locate motifs true motif-containing sequences, as indicated in the second column of tables 5.4 and 5.5, however when one includes the false positive residues predicted to be part of motifs in non-motif containing sequences, the precision drops off substantially, particularly in table 5.5 where it is below both extensions in terms of precision.

5.4 Predictive ability

Given our current biological knowledge about type II secretion in bacteria, Gram negative cytoplasmic proteins would not be expected to have a type II secretion motif. Thus, cytoplasmic proteins form a useful negative testing set, and any cytoplasmic proteins that are predicted from a Gibbs-HMM or GLAM2 model to have a type II secretion motif would be treated as false positives.

Five fold cross validation was done, in which, added to the 1/5 test set of type II secreted proteins, was a large set of cytoplasmic proteins. In cases where cytoplasmic proteins were used to "contaminate" the input training dataset, these proteins were excluded from the

test set.

The model, either the GLAM2 model which is very similar to a profile-HMM, or from our implementations an actual profile-HMM, is taken as output from the training step. Then test sequences are run against the model. The choice of E-value cutoff or score threshold can be somewhat arbitrary. In the case of HMMER, a score of each sequence against the model is returned, and an E-value, the expected number of such sequences expected to align with equal or greater score is returned. After seeing the results and the large number of false cytoplasmic positives included, for each method, the top scoring 25 proteins were counted towards the overall precision/recall tabulations. True motif-containing proteins falling below this list were false negatives, while cytoplasmic and motif-containing proteins on the top 25 list were false positives and true positives respectively.

As indicated in Table 5.6 - 5.8, no model generated from any of the methods effectively classifies cytoplasmic sequences and type II sequences. It should be kept in mind that the ratio of the negatives to the positives in the testing set is approximately 50:1 meaning that if only 2% of the cytoplasmic proteins are included as false positives, we can have only maximum 50% precision, but this situation is unavoidable if we wish to scan large databases of proteins for only a small number of hits. Also of note is the fact that our extended algorithms models decrease in precision/recall slightly less than GLAM2, which was more prone to inclusion of cytoplasmic sequences in the final alignment.

Results are much lower compared to SignalP [7], which is a more hand-crafted approach towards predicting the same type of motifs, however, two points come out:

1) SignalP scans only the N-terminal region, where these motifs are known to exist in nature, here the motifs are planted randomly, and whole protein sequences are scanned.
2) Precision/Recall numbers quoted by SignalP were with a much smaller negative testing set of cytoplasmic proteins than used here, and some of the false positives obtained with the GLAM2 or Gibbs-HMM model are also false positives with SignalP, though at a lower level.

Ultimately, the results reveal that the automatically created models, though able to zero in on the correct motif, are insufficient in their own right as classifiers for this type of data; further handcrafting or adjustment of the model would be needed.

Efforts were also made to attempt the classification problem with the three larger PROSITE-pattern containing datasets, however the problem of defining a truly negative set is something that would require more careful curation. Though a negative testing set

Method	Precision	Recall
Gibbs-HMM	0.64	0.57
GLAM2	0.61	0.51
Gibbs-HMM-Subset	0.59	0.52
Gibbs-HMM-Clustering	0.62	0.58

Table 5.6: Comparison of Predictive Ability for 4 methods to classify sequences correctly based on the presence or absence of a type II secretion motif, when the input dataset consists purely of type II secretion proteins.

Method	Precision	Recall
Gibbs-HMM	0.52	0.47
GLAM2	0.55	0.46
Gibbs-HMM-Subset	0.57	0.50
Gibbs-HMM-Clustering	0.54	0.54

Table 5.7: Comparison of Predictive Ability for 4 methods to classify sequences correctly based on the presence or absence of a type II secretion motif, when the input dataset consists of 80% type II secretion proteins and 20% cytoplasmic (non type-II) proteins.

Method	Precision	Recall
Gibbs-HMM	0.48	0.44
GLAM2	0.47	0.55
Gibbs-HMM-Subset	0.53	0.50
Gibbs-HMM-Clustering	0.53	0.58

Table 5.8: Comparison of Predictive Ability for 4 methods to classify sequences correctly based on the presence or absence of a type II secretion motif, when the input training dataset consists of 67% type II secretion proteins and 33% cytoplasmic (non type-II) proteins.

consisting of proteins that did not match the regular expression was initially formed, it quickly became clear that proteins within this set had matching sub-regions very similar to the regular region but did not give a proper match to the qualitative regular expression. These proteins were emerging as "false positives" with both GLAM2 and Gibbs-HMM, because they were in the negative set despite actually having a region very close to matching the pattern. Due to the abundance of poorly annotated proteins in UniProt, one cannot be sure by casual checking whether a protein is a really a false positive or contains an instance of the motif not exactly matching the regular expression, but has not yet been annotated.

It is a more cut-and-dry situation with type II secretion, where, based on our knowledge of biology, cytoplasmic proteins should not have something closely resembling a type II motif.

5.4.1 Other protein sets attempted

The initial focus of this approach was to build classifiers for protein localization, or for specific, previously uncharacterized secretion motifs such as type III and type IV gram negative bacterial proteins.

Attempts at this have not progressed particularly smoothly, perhaps due to the large diversity of proteins in the relatively small type III and type IV sets. Two large subsets of type III secreted proteins were identified, corresponding to two genera, Chlamydia, and Pseudomonas. These two sets were input independently to GLAM2 and the basic Gibbs-HMM algorithm, motifs found in the Chlamydia set were found to correspond to transmembrane regions. Even though these are secreted proteins, they infect the cells of other organisms, and have regions that span an inclusion membrane with unique chemical characteristics, making them more closely resemble cytoplasmic membrane proteins of Gram negative bacteria. On the other hand, in the Pseudomonas proteins it was more difficult to effectively discover common motifs. Refinements of the dataset are needed, but it may also be simply the case that a higher order structural motif that cannot be effectively represented by a profile-HMM is present in this class of proteins.

Chapter 6

Conclusion and Future Work

6.1 Contributions

In this thesis, we proposed a novel extension of the basic Gibbs Sampling approach to motif discovery, making use of a more complicated motif representation (Profile-HMMs) to allow for insertions and deletions to take place in the multiple alignment of the protein motifs.

Analysis of multiple alignment accuracy using a benchmark dataset of PROSITE pattern-containing proteins was carried out with our new approach, and compared with several existing motif discovery methods, including the very recently published GLAM2 software [18]. Our method, Gibbs-HMM, outperformed all existing methods except for the state-of-the-art GLAM2, developed recently by Timothy Bailey, one of the original developers of MEME [5]. Results were also comparable to GLAM2s performance in terms of locating planted type II secretion motifs in Gram negative bacterial proteins, and in classifying sequences using the motif output from the two approaches.

Gibbs-HMM and GLAM2 make use of different methods for dealing with the problem of a minority of input sequences not containing a motif common to the remainder. In experiments done with three of the PROSITE protein sets and the type II secretion dataset containing proteins with non-motif containing proteins added, Gibbs-HMM outperformed GLAM2 regarding classifier ability from motifs derived from the type II secretion set, performed comparably with two of the PROSITE sets, and underperformed with one PROSITE set. Advantages and disadvantages of both our approach and with GLAM2s approach exist, depending on the type of data involved.

6.2 Further work

Particularly with the recent introduction of the GLAM2 method for gapped motif discovery in biological sequences, there are still many tantalizing paths of future investigation.

Use of additional information beyond the basic amino acid alphabet. Our approach sought to be as generic as possible, and the amino sequence of a protein is generally one of the first pieces of information we have about it. Certain other pieces of information, such as secondary structure and solvent accessibility [2] can be readily obtained from computational prediction. Secondary structure can be represented as a discrete alphabet, with as few as three letters (helix, loop, and coil) superimposed on the amino acid sequence. Solvent accessibility, while a real value that represents the proportion of an amino acids surface areas exposure to the surrounding solvent, can also be discretized to an alphabet. Shared motifs could, in some cases, share common secondary structure or solvent accessibility despite low levels of amino acid sequence conservation. Efforts to incorporate this information, for example combining the 3-letter secondary structure alphabet with the 20-letter amino acid alphabet to form a 60-letter alphabet representation of proteins and discover motifs within these proteins, but were ultimately abandoned due to the challenges presented by uncertainty in the predictive ability of secondary structure/solvent accessibility methods, and the effort needed to carefully craft Dirichlet priors using this expanded alphabet for use with the profile-HMM.

Use of a negative training set. A subclass of motif discovery algorithms [46] take as input both a positive and negative dataset of proteins, with the goal being to find motifs that are overrepresented in the positive set relative to the negative one. To our knowledge, existing methods only use ungapped representations of motifs. The localization problem presents uses of such an algorithm, such as a set of proteins sharing a certain localization and function, and a negative set of proteins with a different localization but similar function, and we wish to find motifs related to localization, rather than the stronger functional motifs. Extending our approach to a discriminative approach would be a future challenge. Mamitsuka [36] presents an interesting approach towards training an HMM using negative training data as well as positive training data, and may be a future starting point for a profile-HMM discovery algorithm that uses negative data during the training stages.

Selection of length criteria. Currently the algorithm is run different times with different choices of motif length, a leave-one-out analysis is done on the scores obtained with the motifs and the top scoring model is chosen. GLAM2's approach of sampling whether or not to add or delete columns to the alignment based on scores obtained from the alignment could be a useful inspiration for a more efficient means of choosing optimal motif length. Gibbs-HMM obtains similar overall alignment precision (PPV) and 15% lower sensitivity than GLAM2, indicating that while correct regions are in general being identified as motifs, the motifs are not always extending to the full proper length of the alignment.

Assumptions made with motif sequences. Dirichlet priors derived for use in HMMER were based on alignments of large, relatively conserved domains. Could more useful priors be derived for alignments of the short protein motifs we are interested in be derived? Would these improve the effectiveness of the approach?

Motif discovery using higher order models than seen here, or at least outputting a final model better able to classify sequences than was encountered with the models formed by both our approach and by GLAM2

Bibliography

[1] Vergunst A.C., van Lier M.C.M., den Dulk-Ras A., Stuve T.A.G., Ouwehand A., and Hooykaas P.J.J. Positive charge is an important feature of the C-terminal transport signal of the VirB/D4 translocated proteins of agrobacterium. *Proceedings of the National Academy of Sciences of the United States of Amierca*, 102:832–837, 2005.

[2] R. Adamczak, A. Porollo, and J. Meller. Combining prediction of secondary structure and solvent accessibility in proteins. *Proteins: Structure, Function and Bioinformatics*, 59:467–75, 2005.

[3] S.F. Althschul et al. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Research*, 25:3389–3402, 1997.

[4] T.K. Attwood et al. PRINTS and its automatic supplement preprints. *Nucleic Acids Research*, 31:400–402, 2003.

[5] T.L. Bailey and C. Elkan. Unsupervised learning of multiple motifs in biopolymer sequences using expectation maximization. *Machine Learning*, 21:51–80, 1995.

[6] A. Bateman et al. The PFAM protein families database. *Nucleic Acids Research*, 30:276–280, 2002.

[7] J.D. Bendsten, H. Nielsen, G. von Heijne, and S. Brunak. Improved prediction of signal peptides: SignalP 3.0. *Journal of Molecular Biology*, 340:783–795, 2004.

[8] J.D. Bendsten, H. Nielsen, D. Widdick, T. Palmer, and S. Brunak. Prediction of twin-arginine signal peptides. *BMC Bioinformatics*, 6:167–176, 2005.

[9] M. Bhasin, A. Garg, and G.P.S. Raghava. PSLpred: prediction of subcellular localization of bacterial proteins. *Bioinformatics*, 21:2522–2524, 2005.

[10] G. Casella and E.I George. Explaining the Gibbs Sampler. *The American Statistician*, 46:167–174, 1992.

[11] P. Christie and E. Cascales. Structural and dynamic properties of the type IV secretion systems. *Molecular Membrane Biology*, 22:51–61, 2005.

[12] N.P. Cianciotto. Type II secretion: a protein secretion system for all seasons. *Trends in Microbiology*, 13:581–588, 2005.

[13] The UniProt Consortium. The universal protein resource (UniProt). *Nucleic Acids Research*, 35:D193–D197, 2007.

[14] P. Delepelaire. Type I secretion in gram-negative bacteria. *Biochimica et Biophysica Acta*, 1694:149–161, 2004.

[15] R. Durbin et al. *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*. Cambridge University Press, 1998.

[16] S.R. Eddy. Profile hidden markov models. *Bioinformatics*, 14:755–763, 1998.

[17] R.C Edgar. MUSCLE: multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Research*, 32:1792–1797, 2004.

[18] M.C. Frith, N.F. Saunders, B. Kobe, and T.L Bailey. Discovering sequence motifs with arbitrary insertions and deletions. *PLoS Computational Biology*, 4:e1000071, 2008.

[19] J. L. Gardy et al. PSORT-B: Improving protein subcellular localization prediction for gram-negative bacteria. *Nucleic Acids Research*, 31:3613–3617, 2003.

[20] J. L. Gardy et al. PSORTb v. 2.0: expanded prediction of bacterial protein subcellular localization and insights gained from comparative proteome analysis. *Bioinformatics*, 21:617–623, 2005.

[21] J.L. Gardy and F.S.L. Brinkman. Methods for predicting bacterial subcellular localization. *Nature Reviews Microbiology*, 4:741–751, 2006.

[22] M Gribskov, A.D. McLachlan, and D Eisenberg. Profile analysis: detection of distantly related proteins. *Proceedings of the National Academy of Sciences of the United States of America*, 84, 1987.

[23] D.H Haft, Selengut J.D., and O. White. The TIGRFAMS database of protein families. *Nucleic Acids Research*, 31:371–373, 2003.

[24] S. Henikoff and J.G. Henikoff. Automated assembly of protein blocks for database searching. *Nucleic Acids Research*, 19:6565–6572, 1991.

[25] S. Henikoff and J.G. Henikoff. Amino acid substitution matrices from protein blocks. *Proceedings of the National Academy of Sciences of the United States of America*, 89:10915–10919, 1992.

[26] S. Henikoff and J.G. Henikoff. Using substitution probabilities to improve position specific scoring matrices. *CABIOS*, 12:135–143, 1996.

[27] N. Hulo et al. The 20 years of PROSITE. *Nucleic Acids Research*, 36:D245–D249, 2008.

[28] I. Jonassen, J.F. Collins, and D. Higgins. Finding flexible patterns in unaligned protein sequences. *Protein Science*, 4:1587–1595, 1995.

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

[30] L. Journet, K.T. Hughes, and G. Cornelis. Type III secretion: a secretory pathway serving both motility and virulence. *Molecular Membrane Biology*, 22:41–50, 2005.

[31] S. Kawashima et al. AAindex: amino acid index database, progress report 2008. *Nucleic Acids Research*, 36, 2008.

[32] C.E. Lawrence, S.F. Altschul, M.S. Boguski, J.S. Liu, A.F. Neuwald, and J.C. Wootton. Detecting subtle sequence signals: A Gibbs Sampling strategy for multiple alignment. *Science*, 262:208–214, 1993.

[33] C.E. Lawrence and A.A. Reilly. An expectation maximization (EM) algorithm for the identification and characterization of common sites in unaligned polymer sequences. *Proteins*, 7:41–51, 1990.

[34] W. Li and A. Godzik. Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences. *Bioinformatics*, 22:1658–1659, 2006.

[35] Z. Lu et al. Predicting subcellular localization of proteins using machine-learned classifiers. *Bioinformatics*, 20:547–556, 2004.

[36] H. Mamitsuka. A learning method of hidden markov models for sequence discrimination. *Journal of Computational Biology*, 3:361–373, 1996.

[37] M. Muller and R.B. Klosgen. The tat pathway in bacteria and chloroplasts. *Molecular Membrane Biology*, 22:113–121, 2005.

[38] K. Nakai. Protein sorting signals and prediction of subcellular localization. *Advances in Protein Chemistry*, 54:277–344, 2000.

[39] A.F Neuwald and J.S. Liu. Gapped alignment of protein sequence motifs through Monte Carlo optimization of a hidden markov model. *BMC Bioinformatics*, 5:157, 2004.

[40] A.F. Neuwald, J.S Liu, and C.E. Lawrence. Gibbs motif sampling: detection of outer membrane protein repeats. *Protein Science*, 4:1618–1632, 1995.

[41] H. Nielsen, J. Engelbrecht, S. Brunak, and G. von Heijne. Identification of prokaryotic and eukaryotic signal peptides and prediction of their cleavage sites. *Protein Engineering*, 10:1–6, 1997.

[42] J. Pei and N.V. Grishin. PROMALS: towards accurate multiple sequence alignments of distantly related proteins. *Bioinformatics*, 23:802–808, 2007.

[43] PROSITE. Zinc finger multiple sequence alignment. http://www.expasy.ch/cgi-bin/aligner?psa=PS00028, Accessed September 3, 2008.

[44] A.P. Pugsley. The complete general secretory pathway in Gram negative bacteria. *Microbiology and Molecular Biology Reviews*, 57:50–108, 1993.

[45] P. Puntervoll et al. ELM server: a new resource for investigating short functional sites in modular eukaryotic proteins. *Nucleic Acids Research*, 31:3625–3630, 2003.

[46] E. Redhead and T.L. Bailey. Discriminative motif discovery in DNA and protein sequences using the DEME algorithm. *BMC Bioinformatics*, 8:385, 2007.

[47] S.K. Sahu and G.O. Roberts. On convergence of the EM algorithm and the gibbs sampler. *Statistics and Computing*, 9:55–64, 1999.

[48] G.K. Sandve and F. Drablos. A survey of motif discovery methods in an integrated framework. *Biology Direct*, 1:11, 2008.

[49] C.J.A Sigrist et al. PROSITE: A documented databaes using patterns and profiles as motif descriptors. *Briefings in Bioinformatics*, 3:265, 2002.

[50] T.F Smith and M.S. Waterman. Identification of common molecular subsequences. *Journal of Molecular Biology*, 147:195–197, 1981.

[51] E.C. Su, H.S. Chiu, A. Lo, J.K. Hwang, T.Y. Sung, and W.L. Hsu. Protein subcellular localization prediction based on compartment-specific features and structure conservation. *BMC Bioinformatics*, 8:330, 2007.

[52] D. Thanassi, C. Stathopoulos, A. Karkal, and H. Li. Protein secretion in the absence of ATP: the autotransporter, two-partner secretion and chaperone/usher pathways of gram-negative bacteria. *Molecular Membrane Biology*, 22:63–72, 2005.

[53] G. Thijs, M. Lescot, K. Marchal, S. Rombauts, B. de Moor, P. Rouzé, and Y Moreau. A higher-order background model improves the detection of promoter regulatory elements by Gibbs sampling. *Bioinformatics*, 17:1113–1122, 2001.

[54] G. Thijs, K. Marchal, M. Lescot, S. Rombauts, B. de Moor, P. Rouzé, and Y. 2001b Moreau. A Gibbs sampling method to detect overrepresented motifs in upstream regions of coexpressed genes. *Journal of Computational Biology*, 9:447–464, 2001.

[55] J.D. Thompson et al. CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Research*, 22:4673–4680, 1994.

[56] Yu et al. Predicting subcellular localization of proteins for gram-negative bacteria by support vector machines based on n-peptide compositions. *Protein Science*, 13:1402–1406, 2004.

[57] Y. Zhai and M.H Saier. The beta-barrel finder (BBF) program, allowing identification of outer membrane beta-barrel proteins encoded within prokaryotic genomes. *Protein Science*, 11:2196–2207, 2002.