12-13-2013 12:00 AM

Computational Molecular Coevolution

Russell J. Dickson
The University of Western Ontario

Supervisor
Dr. Gregory B. Gloor
The University of Western Ontario

Graduate Program in Biochemistry
A thesis submitted in partial fulfillment of the requirements for the degree in Doctor of Philosophy

COMPUTATIONAL MOLECULAR COEVOLUTION

(Thesis format: Integrated Article)

by

Russell Dickson

Graduate Program in Biochemistry

A thesis submitted in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

The School of Graduate and Postdoctoral Studies

The University of Western Ontario

London, Ontario, Canada

Abstract

A major goal in computational biochemistry is to obtain three-dimensional structure information from protein sequence. Coevolution represents a biological mechanism through which structural information can be obtained from a family of protein sequences. Evolutionary relationships within a family of protein sequences are revealed through sequence alignment. Statistical analyses of these sequence alignments reveals positions in the protein family that covary, and thus appear to be dependent on one another throughout the evolution of the protein family. These covarying positions are inferred to be coevolving via one of two biological mechanisms, both of which imply that coevolution is facilitated by inter-residue contact. Thus, high-quality multiple sequence alignments and robust coevolution-inferring statistics can produce structural information from sequence alone. This work characterizes the relationship between coevolution statistics and sequence alignments and highlights the implicit assumptions and caveats associated with coevolutionary inference. An investigation of sequence alignment quality and coevolutionary-inference methods revealed that such methods are very sensitive to the systematic misalignments discovered in public databases. However, repairing the misalignments in such alignments restores the predictive power of coevolution statistics. To overcome the sensitivity to misalignments, two novel coevolution-inferring statistics were developed that show increased contact prediction accuracy, especially in alignments that contain misalignments. These new statistics were developed into a suite of coevolution tools, the MIpToolset. Because systematic misalignments produce a distinctive pattern when analyzed by coevolution-inferring statistics, a new method for detecting systematic misalignments was created to exploit this phenomenon. This new method called "local covariation" was used to analyze publicly-available multiple sequence alignment databases. Local covariation detected putative misalignments in a database designed to benchmark sequence alignment software accuracy. Local covariation was incorporated into a new software tool, LoCo, which displays regions of potential misalignment during alignment editing assists in their correction. This work represents advances in multiple sequence alignment creation and coevolutionary inference.

Keywords: Coevolution, multiple sequence alignment, protein structure prediction, local covariation, protein family curation, Mutual Information

Co-Authorship Statement

Chapter 1 was written with helpful editorial comments from GBG.

Chapter 2 is presently in-press to be published in the journal Methods in Molecular Biology as "Bioinformatics Identification of Coevolving Residues" with my supervisor Gregory B. Gloor (GBG). The workflow was designed with contributions from GBG. GBG provided editorial comments.

Chapter 3 is publicly available on arxiv.org as "The MIp Toolset: an efficient algorithm for calculating Mutual Information in protein alignments" co-authored by GBG, who provided helpful editorial comments.

Chapter 4 is under review at the journal Bioinformatics as "Gambling on Gaps: An Explanation of the Alignment-Coevolution Relationship". It is co-authored by GBG who provided helpful editorial comments and assisted with experimental design for Figure 3.

Chapter 5 is published in the journal PLoS One as "Identifying and Seeing beyond Multiple Sequence Alignment Errors Using Intro-Molecular Protein Covariation". I am the first author of this manuscript with Lindi M. Wahl (LMW), Andrew D. Fernandes (ADF), and GBG as co-authors. The experiment that is featured in Figure 5.8 was designed and conducted by LMW. ADF provided a more formal reformulation of an earlier version of the ΔZp statistic. The new coevolution statistics were created with GBG. Software that was used to conduct experiments for this chapter was co-written by GBG. The manuscript was written predominantly by me with GBG. Supplementary figures and data for Chapter 5 is available at plosone.org.

Chapter 6 is published in the journal PLoS One as "Protein Sequence Alignment Analysis by Local Covariation: Coevolution Statistics Detect Benchmark Alignment Errors". It is co-authored by GBG who provided helpful editorial commentary.

Chapter 7 was written with helpful editorial comments from GBG.

Acknowlegements

Foremost, I would like to thank my supervisor, Dr. Gregory Gloor for his guidance, support, patience, and help throughout my studies. I would not be where I am today if it was not for his generous investment of time and effort. I learned a great deal about what it means to be a Professor from him. Being his teaching assistant was an eye opening experience for how to provide an ideal learning environment for students. I also learned the importance scientific and intellectual integrity. I truly appreciate the time I had to learn from him and was a part of his group.

I also wish to thank my thesis advisors Dr. Stan Dunn and Dr. Chris Brandl. During my time at Western I had the pleasure of taking classes with both of these esteemed professors and learned a great deal from both. Notably, a fateful conversation with Dr. Dunn after a 3rd-year biochemistry class led me to the topic of this PhD thesis and Dr. Gloor's lab. I appreciate them taking the time to contribute so profoundly to my graduate education.

Also I must thank members of the Gloor lab, past and present. Especially (future Drs.) Jean Macklaim, Tom McMurrough, and Dr. Andrew Fernandes, whose hard work and friendship are very much appreciated. As well, thanks to Dr. Ardeshir Goliaei from Dr. Dunn's lab. It has been a pleasure working with all of you.

Thanks also to Dr. David Edgell for his mentoring, guidance and support, Dr. Lindi Wahl for her thought provoking insights, and Dr. Mark Daley for his motivating conversations and advice. I deeply appreciate the time that these mentors invested in me.

I would also like to thank Dr. Heather Gordon at Brock University for my first exposure to Bioinformatics in high school that started me on a path to this PhD thesis.

I want to thank my family, especially my parents for their love and support, and for fostering in me a love of science. And to my friends, who endured my attempts to explain coevolution at social gatherings.

Most importantly, I want to thank my loving wife, Jackie. Her love, support, and sense of

teamwork is what kept me going. I couldn't ask for a better, more patient, more understanding, or more supportive partner than her.

I have been supported by an NSERC PGS-M and an NSERC CGS-D scholarships, as well as from The University of Western Ontario, and an NSERC Discovery Grant to Gregory Gloor.

Contents

List of Figures

List of Appendices

List of Abbreviations, Symbols, and Nomenclature

a priori: from an understanding of mechanism rather than empirical observation.

APC: Applied Product Correction. Used to compute *MIp* from *MI*.

BAliBASE: Benchmark Alignment dataBASE. A collection of sequence alignments for evaluating sequence alignment methods.

bHLH: basic Helix Loop Helix. A family of proteins.

BLOSUM: BLOcks SUbstitution Matrix. A scoring matrix used in sequence alignment.

CAPS: Coevolution Analysis using Protein Sequences. A coevolutionary-inference method.

CLUSTALW: Progressive multiple sequence alignment software.

CMA: Correlated Mutation Analysis. A coevolutionary-inference method.

Cn3D: "See in 3D". Software for viewing and generating structure-based sequence alignments.

covarion: Concomitantly variable codon.

DCA: Direct Coupling Analysis. A coevolutionary-inference method.

de novo: Starting from first principles.

ELSC: Explicit Likelihood of Subset Covariation. A coevolutionary-inference method.

GBLOCKS: Software that removes putatively-unreliable positions from a protein alignment prior to phylogenetic inference.

H: Information entropy, a measure of the uncertainty of a random variable.

HMMER: A Hidden Markov Model-based tool for inferring homology.

LAGLIDADG HE: A Homing Endonuclease protein family.

LDH: Lactate Dehydrogenase. A protein family.

LoCo: Local Covariation-based sequence alignment curation tool.

MAFFT: Multiple Alignment using Fast Fourier Transform. A global sequence alignment method.

McBASC: McLachlan-BAsed Substitution Correlation. A coevolutionary-inference method.

MDH: Malate Dehydrogenase. A protein family.

MIp: product-corrected Mutual Information. A coevolutionary inference method.

MIpToolset: Software for calculating Mutual Information-based coevolution statistics from protein family alignments.

MSA: Multiple Sequence Alignment.

MUSCLE: A global sequence alignment tool.

OMES: Observed Minus Expected Scores. A coevolutionary-inference method.

OXBENCH: A database for benchmarking sequence alignment methods.

PAM: Point Accepted Mutation. A type of substitution scoring matrix for global alignment.

PRANK: A sequence alignment tool that is phylogeny-aware.

PREFAB: A database for benchmarking sequence alignment methods.

PSI-BLAST: A tool for inferring sequence homology.

PSSM: Position-Specific Scoring Matrix. A substitution scoring matrix applied uniquely to each position in an alignment.

SABMARK: A database for benchmarking sequence alignment methods.

SCA: Statistical Coupling Analysis. A tool for inferring coevolution.

SP: Sum of Pairs. A scoring method for evaluating a multiple sequence alignment.

T-COFFEE: A sequence alignment method.

TIM: Triosephosphate Isomerase. A protein family.

VAST: A protein structure alignment method.

Zp: The MIp coevolution statistic converted into Z-scores.

Zpx: A coevolutionary statistic based on Zp.

ΔZp: A coevolutionary statistic based on Zp.

Chapter 1

General Introduction

1.1 Open Problems in Bioinformatics

Intra-molecular protein coevolution represents the confluence of the two largest open problems in bioinformatics, multiple sequence alignment and protein structure prediction. Protein structure prediction has long been a major goal of bioinformatics because of the utility of accurate three-dimensional protein structure models and the comparatively high financial and time investment required to generate such models using *in vitro* rather than *in silico* methods. Coevolution represents a biological mechanism through which multiple sequence alignments can generate structural information.

It is theoretically possible (yet, at present, computationally intractable) to generate a full, accurate three-dimensional model of a protein using only the primary amino acid sequence from which the protein is composed [6]. Because of the computational challenges associated with protein structure prediction, many groups have attempted to limit the massive protein structure prediction "search space" with supplemental data too coarse for accurate modelling by itself. These coarse experimental methods include the incorporation of cryoelectron microscopy and partial nuclear magnetic resonance data [1]. As well, sequence analysis-based methods generate predictions based on models of evolution and multiple sequence alignments

[20].

The potential to produce accurate predictions of the three-dimensional structure of proteins is one example of the vast utility of Multiple Sequence Alignment (MSA). The ability to generate rapid and accurate sequence alignments is one of the largest contributions of the field of computational biology. When conducted properly, sequence alignments identify homology at the molecular level, and thus are possibly the most clear and explicit piece of evidence of evolution itself. Therefore, multiple sequence alignment is fundamental to all of molecular biology as it is a way of exploring homology.

1.2 Protein Molecular Evolution

1.2.1 Homology

Homology is the similarity we see in the natural world that is due to the common descent of organisms; it is one of the most fundamental and important concepts in biology [17, 41]. Two traits are said to be homologous if they show similarity due to ancestral relatedness. If one was to use traits as evidence of relatedness between two organisms, homologous traits would be examples of "signal".

But not all biological similarities can be attributed to shared ancestry; for example, a bat wing and bird wing appear to share superficial similarity in structure, but arose independently. Bat and bird wings are only homologous as the forelimbs of tetrapods, but not as wing structures themselves [70]. The similarity between bird and bat wings is an example of the convergence to a similar solution and is called analogy. If homologous traits are metaphorical "evolutionary signal" as outlined previously, traits that are analogous but not homologous would be "noise".

One of the greatest insights in the history of science was that the history of all living organisms could be represented as a phylogenetic tree with extant organisms represented as terminal leaf nodes, extinct ancestral organisms at intermediate nodes connecting extant organisms, and

a universal common ancestor residing at the root of the tree [17]. A phylogenetic tree is inferred by identifying shared characteristics that define related groups of organisms called synapomorphies and symplesiomorphies. The distinction between the two classes is that symplesiomorphies refer to a shared ancestral character state, and synapomorphies refer to a shared derived character state. These two phylogenetic terms are contrasted by homoplasy, character states whose similarity exists because of convergent evolution. Thus, homoplasies are analogous, but not homologous.

These concepts are often discussed at the organism level, but also apply at the protein level. The evolutionary history of a protein can be represented by a phylogenetic tree and the amino acid identity at the various positions within the protein represent character states that can be synapomorphic, symplesiomorphic, or homoplasic. The phylogenetic tree representing a protein may differ greatly from the phylogenetic tree of its host organism. For example, in the case of horizontal gene transfer between bacteria, the phylogenetic tree of the protein encoded by the gene would more closely resemble that of the transferring organism than it would the recipient organism. These are called xenologous proteins, homologous proteins related by a horizontal transfer[49]. The type of homology between proteins affects the inference of intra-protein coevolution, as outlined below.

1.2.2 Orthology and paralogy

The concept of homology can be subdivided further as we discuss protein molecular evolution. Homologous proteins are similar because of shared ancestry, but this distinction is not sufficient to generate high-quality coevolutionary data. The relationship between the homologous proteins is important and can be divided into several subtypes [49]. Orthologous proteins are related by direct linear descent from a common ancestral protein that were separated by speciation events. Paralogous proteins are proteins that have been separated by a gene duplication where multiple copies of a protein exist in the same genome [29]. Unlike orthologous proteins, where a conservation of function is likely [26], paralogous proteins are likely to undergo

functional divergence over evolutionary time [29]. It is important to remember that both orthology and paralogy are subtypes of homology and can be distinguished from analogy, which is relatedness because of common function.

While it is algorithmically possible to infer gene duplication and speciation events from a complete gene tree [83], there exists no automated way to determine whether two sequences are orthologues or paralogues *a priori*, given only their sequences. However, one major goal of sequence alignment is to create a protein family consisting of orthologous proteins. Thus one challenge of sequence collection through local alignment is to distinguish orthologues from paralogues. There are numerous reasons why a protein family should ideally contain only orthologues depending on the perspective of the user of the protein family. The phylogenetic tree of an orthologous protein family closely resembles the topology of the species tree by definition, and thus the orthologous character of a protein family is useful for phylogenetics; in fact, the consensus of multiple putatively-orthologous protein family trees is the basis for the Tree of Life project [12]. From a functional perspective, it is logical to classify proteins by their utility. The conservation of function between orthologues combined with their shared ancestry makes them colloquially the "same" protein in two different species. Orthologous proteins with ancient origin typically show a conservation of function across the tree of life [26]. When trying to make structural and functional inferences about a specific protein based on the information in a protein family alignment, it is important to include only orthologues because the structural and functional divergence that occurs as the paralogue acquires new function would interfere with the ability to obtain information from regarding the orthologous family of interest.

By contrast, paralogous sequences are the source of novel function because the presence of a second copy of the protein provides the evolutionary freedom to explore new sequence space without interfering with the function of the original orthologous protein. Superfamilies are created when several paralogous protein families are combined together.

1.2.3 Protein families and superfamilies

As a practical example, malate dehydrogenase (MDH) is an enzyme that is part of the citric acid cycle. There are many difference specific sequences from as many organisms that we consider to be part of the MDH family. Based on sequential, structural, and functional analysis, we know that malate dehydrogenase and lactate dehydrogenase (LDH) are homologous and thus we can infer that the two families are paralogous, originating from an ancient gene duplication event [59]. Thus the Conserved Domain Database contains a superfamily that contains a family called "LDH_MDH_like", containing all paralogous members of both families [59]. The decision to include a sequence in an orthologous protein family must ultimately reside with the expert opinion of a curator using evidence like sequence similarity, the number of putatively homologous sequences found in a search, functional annotation, experiment, etc.

The important concept of homology and analogy extends to both the sequence and structure of the protein. Structural and sequential motifs or even positions can be said to be homologous or analogous. However, sequential misalignments are commonly based on the alignment of two analogous sequence motifs, or even analogous residues. Two protein structures may be homologous, as the two folds may be super-imposable for many structurally-conserved features. Structural analogy exists but is comparatively very rare [11].

1.3 Multiple Sequence Alignments

1.3.1 Sequence alignment overview

The concept of homology is the fundamental theoretical underpinning of multiple sequence alignment. DNA and protein sequences are biological traits, just like skeletal wing structure. The major advantage of using biological sequences over other morphological traits is that they are universal and thus can be used to make evolutionary inferences across all known organisms.

In a multiple sequence alignment, each position in the biological sequence, meaning either

a nucleotide for a DNA alignment or amino acid for protein, is aligned into a putatively homologous column [19]. Figure 1.1 shows a graphical representation of an alignment for the protein family triosephosphate isomerase. Residues in this alignment are coloured by amino acid property where similar colours represent similar properties, revealing homologous columns visually [74]. The goal of multiple sequence alignment is to assign each biological character in the sequence to these homologous positions such that the similarity between sequence positions in that column, often manifesting as similar identity or properties, is due to the shared ancestry of the sequences. However, there is the potential for much analogy on a position-by-position basis in sequence alignment because biological characters (nucleotides or amino acids) occur many times and therefore there is often low information content relative to the number of hypotheses tested.

Ideally, a multiple sequence alignment is a grid where every row is a biological sequence and every column is a homologous position. This problem is trivial when aligning near-identical sequences with no insertions or deletions; such a case exists when creating alignments of orthologous sequences from closely-related organisms. However, such ideal conditions are impossible to meet when sequence similarity diverges. When building a protein family model, the goal is to collect all members of the protein family spanning the entire tree of life. This means that the protein family model contains unknown ancient, extinct protein members that cannot be sequenced.

1.3.2 Challenges in multiple sequence alignment evaluation

Unfortunately, there is no deterministic experimental alignment method that provides a definitive solution against which computational solutions can be benchmarked. For a comparison, consider the *in silico* structure prediction problem: Structure predictions generated *in silico* can be compared to structures generated by an established *in vitro* method like X-Ray Crystallography or Nuclear Magnetic Resonance to determine the accuracy of the *in silico* method, and to guide the development of the software [62]. Conversely, there is no *in vitro* experiment

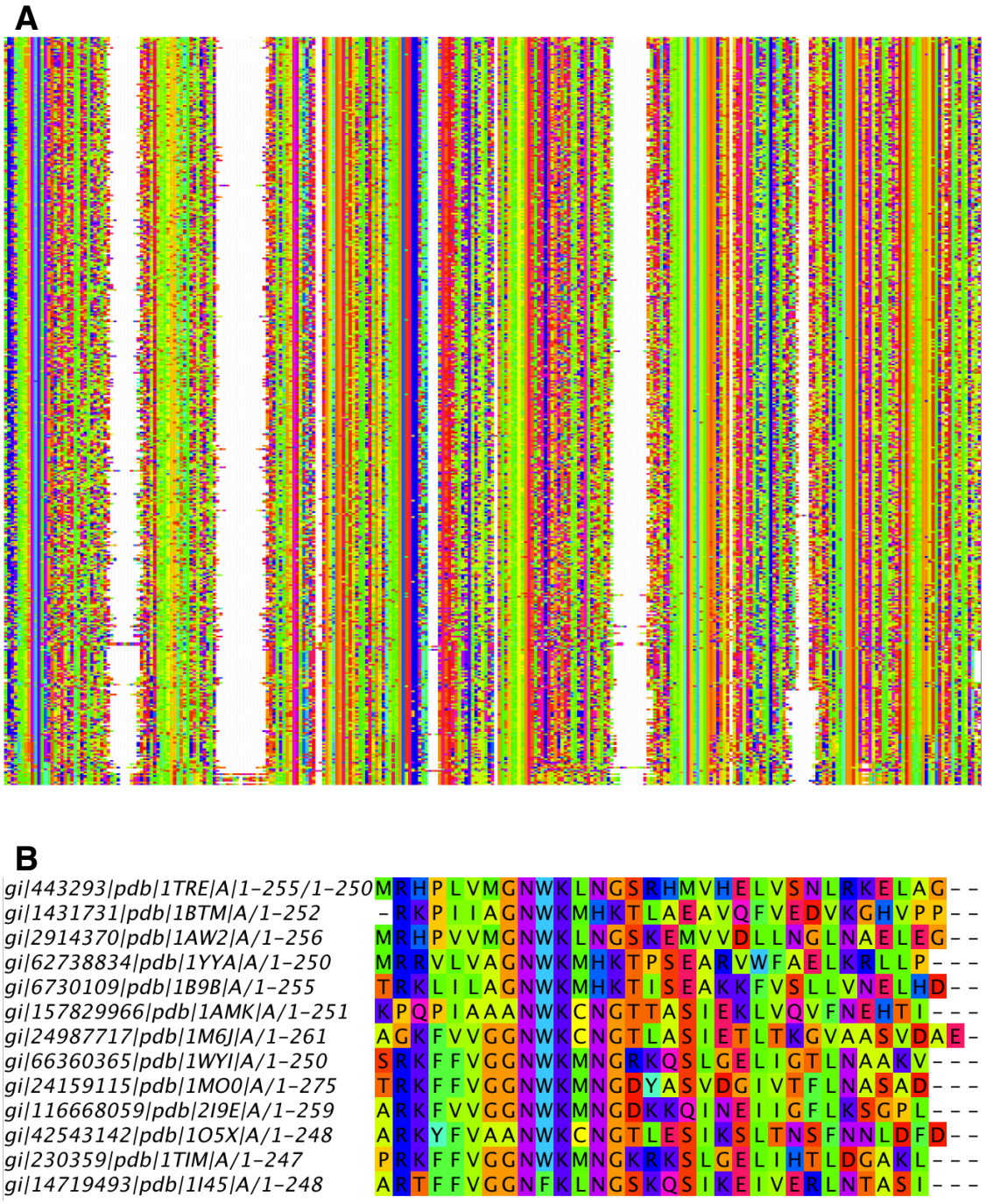

Figure 1.1: Alignment of triosephosphate isomerase created using Cn3D [40] and drawn using Jalview [80]. Amino acids are coloured by amino acid property according to the Taylor colouring scheme [74]. Panel A shows the full alignment. Panel B shows the first contiguous segment for the first 13 sequences with gi and pdb entries visible.

that can generate the "correct" protein multiple sequence alignment because such a hypothetical experiment would involve actually observing mutational and speciation events that have already occurred over the entire history of life on earth. Only extant data can be collected and extinct sequence data is lost to history. While there are methods that attempt to infer the identity of extinct ancestral sequences [65, 38], these methods are hypothesis-generating just like sequence alignment methods.

Robert Edgar, the creator of the widely-used MUSCLE sequence alignment tool [25], publicly declared Multiple Sequence Alignment a "dead field" in 2010. He credited the stagnation of the field to the fact that new ideas in sequence alignment didn't seem to produce noticeable improvements in benchmark datasets. The publication of the original BAliBASE dataset in 1999 [76], of which there are several iterations [9, 77], catalyzed a decade of advancement, as competing groups published work demonstrating that their method provided the highest benchmark scores. BAliBASE was a dataset of carefully constructed alignments that were curated by hand and were structurally-supported; this careful human intervention is what gave confidence in BAliBASE's correctness.

However, there is still no physical experiment that establishes the correctness of BAliBASE or other "benchmark-quality" datasets and thus a potential explanation for the stagnation of the alignment field is that the benchmark alignments are not correct. While the assumed correctness of BAliBASE was necessary for the field to proceed at the inception of the database, it may ultimately have lead to the stagnation of the field a decade later; perhaps the lack of progress in alignment accuracy was due to errors in the database rather than errors in the alignment procedures themselves.

1.3.3 Alignment scoring matrices

The simplest sequence alignment problem involves only two sequences. The algorithmic goal of the two-sequence problem is to maximize a scoring function that is designed with the goal of producing the most-likely correct alignment when the score is highest. A dynamic pro-

gramming approach is used to reuse calculated information efficiently [63]. Matched (or mismatched) residues are given a score according to values in a matrix based on the observable frequency with which such substitutions are seen in alignments relative to the expected frequency background substitutions based on independent positional evaluation [19].

Insertions and deletions are accounted for by inserting one or more "gap characters", typically a dash, and assessing an affine gap penalty, that acknowledges that a single insertion or deletion event can cover many alignment positions and thus penalizes the creation of a gap more strongly than its extension [2]. The first multiple sequence alignments were necessarily made by hand and involved closely-related sequences [19]. Closely-related sequences are said to have high identity because there are few sequence differences between them. Regions that show high identity are said to show conservation, a measure of the extent to which identical amino acids appear in the columns in the alignment [68]. Such hand-made alignments of similar homologous sequences were the basis of the Point Accepted Mutation (PAM) scoring matrices [19]. Entries in a substitution matrix are scaled log-odds scores. They are calculated by taking the log ratio of the empirically observed amino acid substitution frequencies versus the observed marginal frequency of each amino acid, ie. the probability the substitution occurred by accepted mutations versus the probability the amino acids were aligned by chance. If the substitution is empirically favourable, a positive score is assigned to the substitution. PAM matrices are ideal for global alignments of known homologues because of their end-to-end nature and the largely highly conserved alignments due to sequence similarity.

The BLOSUM matrices, developed later, were based on an analysis of more diverged proteins than the PAM matrices [39]. Henikoff et al. analyzed short, highly-conserved segments called blocks and observed amino acid substitution rates. BLOSUM matrices are developed to infer homology between two dissimilar sequences, which is in contrast to PAM matrices that are used to align sequences for which homology has been asserted. Henikoff et al. developed a method to encourage inference between dissimilar sequences. All sequences that are within a defined identity threshold are combined and treated as a single sequence for the purposes of

calculating positional frequencies and accepted mutations.

A scoring matrix provides an empirical method for evaluating more distantly-related sequences. The initial hand-made alignments were based on the concept of identity, where columns were aligned such that as many matching residues were super-imposed in the same column as possible [19]. With the inclusion of scoring matrices, alignments could be measured by similarity, where empirically-derived favourable mutations could contribute positively to the alignment. A completed alignment is usually evaluated based on conservation. Two protein sequences with high identity show conservation. Protein sequences with low similarity show little conservation. Typically, a multiple sequence alignment will show varying degrees of conservation across its length. The most conserved regions are inferred to be important because something is fighting the natural tendency for sequences to diverge in this region [10, 69].

Both PAM and BLOSUM are designed to be general-purpose scoring solutions for their respective problems. However, a better solution can be built for a specific protein family. A Position-Specific Scoring Matrix (PSSM) is similar to the aforementioned scoring matrices except it is calculated on a position-by-position basis for the specific protein family being analyzed [5]. As new sequences are added to a growing alignment, the PSSM is recalculated for each position, expanding the accepted mutations for each position and increasing the likelihood of inferring homology between divergent sequences. A PSSM is an effective alignment tool because it will empirically derive the underlying "rules" of accepted substitutions for each position in a protein family. Whereas a BLOSUM matrix may suggest that two amino acids are fairly interchangeable in the general case, the same substitution may not be allowed at all at a specific position in the protein family being analyzed. If a PSSM is built from a diverse enough collection of sequences, it can be a powerful tool for inferring homology between diverged sequences.

1.3.4 Local and global alignment

The distinction must be made between local and global alignment, with the former being designed for finding putatively homologous sequences, and the latter for creating an end-to-end alignment that encompasses the evolutionary relationships of a protein family only once homology between all sequences can be assumed.

Local alignment is used for sequence collection, since no *a priori* assumption of homology is made. The most common tool, BLAST, uses matrices and affine gap scores, but accelerates search speed by pre-processing its database in order to increase search-time speed [3]. PSI-BLAST is an iterative approach to local alignment with increased accuracy in collecting distantly-related sequences [5, 4]. In PSI-BLAST, Position-Specific Scoring Matrices (PSSM) are used to increase sensitivity to the empirically observed substitution frequencies at each position in the protein individually. Each position in the growing alignment has its own empirically-derived scoring matrix that is recalculated as new putatively-homologous sequences are added to the growing alignment. The presence of an intermediate sequence between two dissimilar members of the same protein family will improve the algorithm's ability to assert homology between the three.

Because PSI-BLAST only uses affine gap penalties [2] for insertion and deletions placement, some choose to use Hidden Markov Model-based tools like HMMER [28, 23], which allows for a more sophisticated gap model at the cost of complexity of the alignment model. A Markov Model is a graph such that each node is a state and each edge contains a probability that the edge will be traversed. A Hidden Markov Model is a Markov Model where the current state is unknown. A Hidden Markov Model can be used to generate an alignment by determining the alignment with the highest probability according to the model.

Once sequences are collected by local alignment, they are built into a protein family model commonly called a multiple sequence alignment (MSA) by using global alignment. Local and global strategies differ algorithmically in their choice of scoring matrix, their affine gap penalties, and the treatment of negative numbers when extending an alignment. But in a MSA

algorithm, the most important issue is in what order, and in what way, each sequence is aligned to the rest.

1.3.5 Global multiple sequence alignment methods

Sequence-based alignment methods are distinguished from structure-based multiple sequence alignment methods in that they are "all-to-all" methods, where there is no obvious sequence or group of sequences that are considered more important *a priori*. Conversely, structure-based alignment is a "master and slave" strategy, where new sequences are aligned to a "master" structure alignment that is based on structural rather than sequential similarity. The sum of pairs (SP) scoring function involves summing all the pairwise alignment scores implied by the alignment, and is the simplest method for evaluating a multiple sequence alignment. While simple, it is computationally intractable to find a near-optimal solution in this search space using a global search strategy, as there are too many possible alignments to evaluate each one. A strategy is necessary to generate an alignment in reasonable time.

The most common strategy is the progressive strategy, used by CLUSTALW [75] and T-COFFEE [64]. In progressive alignment, sequences are evaluated by a precomputed similarity function and aligned in order such that the most similar sequences are aligned together first. The order that the sequences will be aligned forms a guide tree that controls the flow of the alignment process. As the algorithm continues, more sequences are added sequentially to the growing alignment until all the sequences have been added to the alignment.

This strategy has the major drawback that early errors are propagated as the alignment grows. Sometimes two alternative alignment choices score equally well and thus the algorithm selects one of the two alternatives *arbitrarily*. If such a decision is made incorrectly due to insufficient information early in the alignment process, new sequences added to the alignment may also be aligned incorrectly and a systematic misalignment is produced.

Early choices have significant impact on the final alignment, which is why similar sequences are aligned first in the hope that they will yield the fewest errors. The iterative align-

ment, used by MUSCLE [25] and MAFFT [46, 45], strategy was designed to compensate for systematic errors by adding an iterative refinement step: After the progressive strategy has produced an alignment, the iterative strategy searches for alternative alignments that may have a higher optimum score than the current progressive alignment. This post-processing step can eliminate some of the systematic errors generated by the progressive strategy.

Finally, a more sophisticated sequence-based alignment method is the phylogeny-aware strategy used by PRANK [57, 58]. This method attempts to infer synapomorphic insertions and deletions. Phylogeny-aware algorithms use an adjoining phylogenetic tree to attempt to align homologous residues in gap regions. This results in much longer alignments since a gap may be composed of multiple independent insertion or deletion events. The important corollary to this observation is that most alignment methods that do not incorporate a phylogeny aware strategy will maximize alignment scores by aligning homoplasic positions especially in gap regions.

1.3.6 Structure-guided multiple sequence alignment

One of the most effective strategies for improving MSA quality is structure-guided alignment. Members of the same protein family show strong conservation of structure in the catalytic core. Figure 1.2 is a structure alignment of triosephosphate isomerase that shows the strong structural superimposition of the protein backbone through the core of the protein family. It is possible to infer homology between very sequentially-dissimilar proteins using structural information [42]. Thus structure is a powerful tool for creating alignments in diverse protein families. Structure-based strategies involve collecting all members of a protein family for which there is structural data and super-imposing each structurally-homologous residue.

The VAST structure alignment tool [32] is notable because it serves as the structure alignment software for the Cn3D alignment tool [40] that is used to curate the Conserved Domain Database [59]. VAST is able to rapidly identify structural homology and align protein structures by abstracting a protein structure into major structural elements, helices and sheets, in

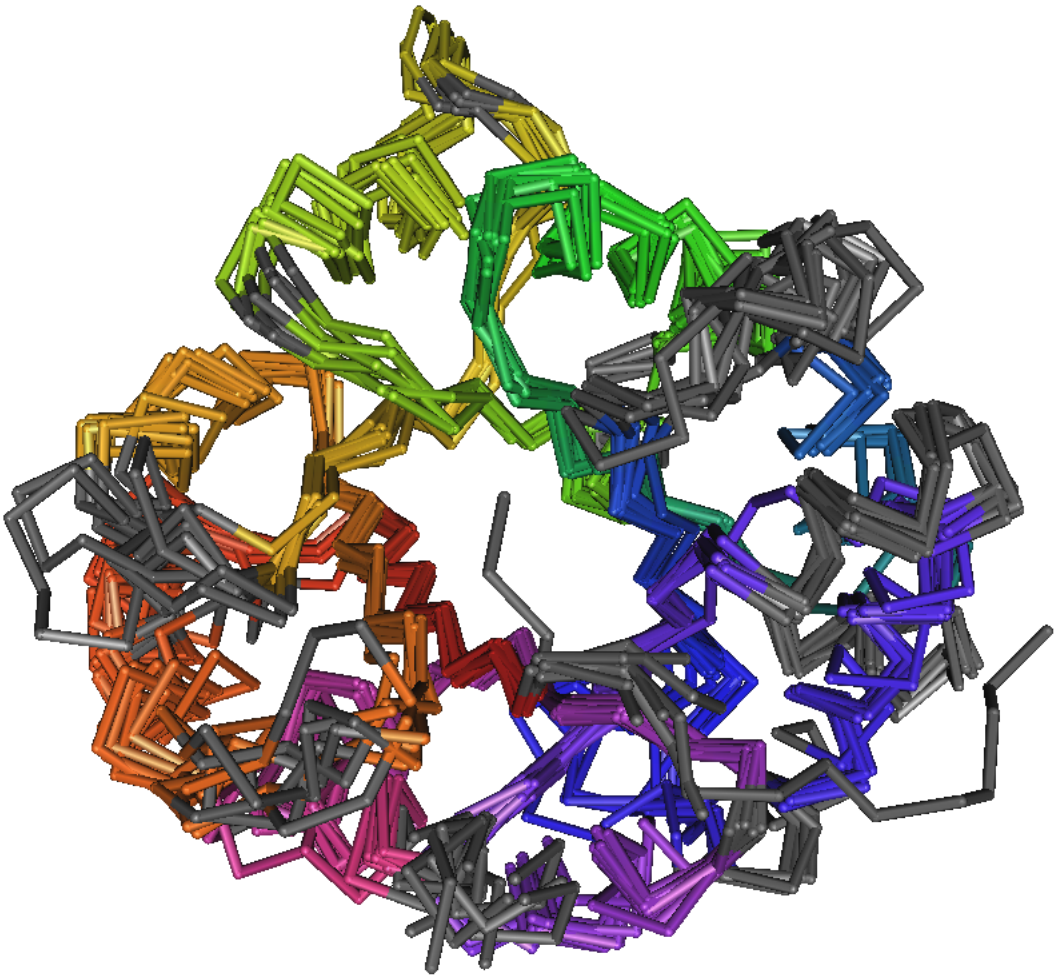

Figure 1.2: Structure alignment of triosephosphate isomerase created and rendered in Cn3D [40]. Aligned regions are coloured by the rainbow colouring scheme to assist in visualizing the contiguous protein chain. Structurally divergent regions are coloured grey and are not aligned.

order to reduce the search space. VAST attempts to align the conserved core secondary structural elements and ignores unstructured surface loops. The structure alignment can be used to generate a sequence alignment by analyzing inter-α-carbon distances.

This abstraction used to create the structure alignment draws attention to the issue of shift

errors [47]. The structure alignment may largely superimpose two secondary structural elements, but the α-carbon superimposition may incorrectly imply a sequence alignment that is shifted one or more residues towards the N- or C-terminus. From a structural inference perspective, these errors are minor and fairly acceptable; from a sequence alignment perspective, it is a serious error that can be difficult to detect in downstream analyses.

The inferred sequence alignment is the basis for a growing sequence alignment [40]. A PSSM is calculated from the structurally conserved regions and it is used to align new sequences to the structure alignment. As each sequence is added to the original structure alignment, the PSSM is recalculated allowing for greater confidence when aligning dissimilar sequences. When a diverse protein family has multiple structures from divergent organisms, it is possible for the PSSM to be calculated such that the diversity of the alignment can be taken into account and a high quality alignment is made from sequences that would not be alignable from sequence alone. This strategy is, as mentioned, "master and slave" because new sequences are aligned to an established structure-based alignment that is assumed to be correct.

When done correctly, structure-based alignments are considered to be of the highest quality [76]. However, great care must be taken to align the structures carefully, because structure alignment methods place a much higher emphasis on this original alignment than sequence alignment methods, which should function more as an all-to-all method. As well, structure alignment is hampered by the fact that structure is comparatively rare and is expensive to produce. Furthermore, using structure alignment is not possible if the protein alignment is being made to create protein structure predictions.

While it is typical to benchmark coevolution methods using structure-based alignments, because structure is necessary to test contact prediction accuracy and structure-based alignments are generally considered to be of optimal quality, such alignments are clearly not viable for use in coevolution-based structure prediction analyses. The presence of a homologous structure would make homology modelling the most logical strategy for *in silico* structure prediction, and make coevolution unnecessary for this purpose. However, the presence of structure data

in a high quality structure-based multiple sequence alignment does not preclude coevolution analysis for obtaining novel insights about residue positions important to the protein's structure, function, or evolution.

1.3.7 The role of gaps in an alignment

Sequence alignment methods can be split into two broad categories of which there are many subtypes: those that use additional structural information and those that use only sequence [71]. The differences between these two methods illuminate a schism in the field on the definition of what exactly constitutes a "homologous position", which residue in a particular sequence belongs in that position, and what the exact definition of a gap should be. This disagreement over such minutiae emphasizes the importance of accurate benchmark alignment databases for their ability to focus the field on optimizing software development for use on a putatively correct set of hand-curated, structure-based alignments. From a structural perspective, gap characters are an alignment artifact.

Figure 1.3 demonstrates gaps from a structural perspective. The surface loop depicted in Figure 1.3 (grey positions) represents multiple different solutions to the problem of connecting two structurally-conserved segments of the protein (coloured positions). The structural superimposition of the α-helix and β-sheet regions is apparent and thus the inferred sequence alignment from these regions is clearly defined. Conversely, the loop that connects these two regions diverges. The shortest segment connecting the two structured regions is three amino acids long, and the longest is ten. A subset of the structures may show structural similarity in this region, but there is not a structurally-homologous position defined for the entire protein family for any position in this region. A sequence alignment program may align amino acids corresponding to the structural positions shown based on a substitution model, but such an alignment is not clearly defined by structural similarity. The documentation for the Cn3D structure alignment tool explicitly states that these structurally unaligned amino acids will appear in the sequence alignment as a convenience to the users, but that "nothing can nor should

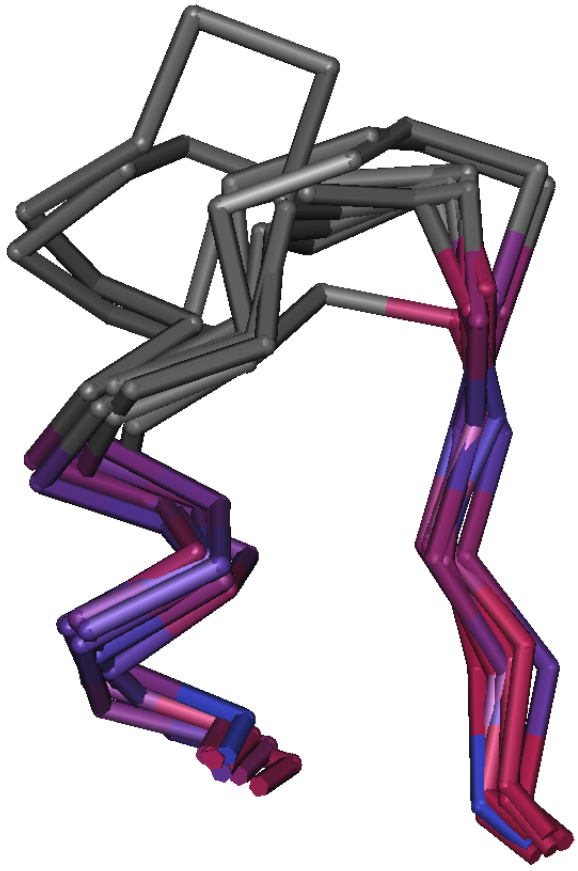

Figure 1.3: Structure alignment of a structurally-divergent surface loop (grey) connecting a structurally-conserved α-helix and β-sheet (coloured) in triosephosphate isomerase. The alignment was created and rendered in Cn3D [40]. The structurally-conserved regions are coloured according to the fit colour scheme. The surface loop region is between three and ten amino acids long.

be inferred from the apparent 'alignment' of residues in the unaligned areas" [14].

Structurally, a gap character literally represents "nothing" since there are no positional coordinates that correspond with the gap position. Thus, methods attempting to infer structural information from a protein family alignment will often ignore the often-unstructured gap regions [32]. Conversely, to a sequence-based approach a gap can appear meaningful. To a purely computational analysis, the gap character is simply another character and will be included as though it were the 21st amino acid [54, 55, 53, 72, 61]. For phylogenetics analysis, gaps do

contain information, but they cannot be treated as another amino acid [37, 43]. As mentioned, affine gap scores are used because a single insertion or deletion even can create multiple "gap characters" in the alignment. The presence or absence, and length of a gap do contain phylogenetic information. However, it is common practice to discard all positions containing gaps before building a phylogenetic tree [37, 43, 73].

1.3.8 Assessing alignment quality

The strategies used by phylogenetics methods when analyzing alignments may be instructive. GBLOCKS is designed to assess whether a section of an alignment should be included in the calculations to generate a phylogenetic tree [73]. GBLOCKS assesses the conservation of a position to determine whether it may contain errors and then determines, based on user parameters, whether the position should be kept in the analysis. By default, GBLOCKS rejects positions containing gap characters. It is worth noting that conservation is used to generate the alignment and to filter which positions should be kept by GBLOCKS. Thus, systematic errors that occur in the alignment process may not be detected by GBLOCKS or other conservation-based methods as they have already been validated by the same criterion earlier in the analysis pipeline.

A major challenge in sequence alignment is the lack of a "correct answer" which can be experimentally derived *in vivo* or *in vitro*. Unlike *in silico* structure prediction, which can benchmark against crystal structures, there is no equivalent experiment that can generate a benchmark dataset. In order to generate such a database by experiment, one would need to have observed every sequence alteration when it occurred in species that are separated by evolutionary time scales. It is therefore tempting to generate synthetic data on which to benchmark one's alignment generation and analysis methods. This strategy must be approached with caution as one's synthetic alignment database will be evolved using the same assumptions that the sequence alignment method uses. This will result in greatly overestimating the alignment method's accuracy on biological data. It is crucial to benchmark alignment analysis methods

against real biological sequences especially from public databases.

There are now many benchmark from which one can obtain demonstration alignments including BAliBASE 3 [77], OXBENCH [67], PREFAB [25], and SABMARK [79]. These benchmarks rely on structural alignments to assert the correctness of the sequence alignment. However, structure alignments are known for containing shift errors and thus are not completely reliable [47]. As well, structure alignment algorithms often disagree [36], making it impossible to guarantee correctness when using structure alignment as a validation criteria. Robert Edgar highlighted inconsistencies in major protein alignment benchmarks [24]. According to Edgar, SABMARK superfamily alignments contain self-inconsistent placement of sequences, and BAliBASE contained alignments of non-homologous domains, and groups of homologous sequences aligned unnecessarily to gaps rather than each other.

1.4 Intra-Molecular Protein Coevolution

1.4.1 Positional non-independence

A common assumption made by sequence analysis tools is that positions within a biological sequence are independent, despite the overwhelming evidence that they are not [16, 82, 30, 66, 81]. The precise model of sequence change within a protein is still unknown and it likely lies somewhere on a continuum between two models: The second-site suppression model [82], and the concomitantly variable codon model (covarion model) [30]. These models were created while investigating the mutability of various protein families.

During work to establish the genetic code, Yanofsky et al. noted that mutations in tryptophan synthetase could be suppressed by changes at a second site [82]. A pattern of changes emerged wherein the second site suppressors seemed to occur at specific locations. Yanofsky inferred that, though they were distant in sequence, they were close in three-dimensional space. Thus, the second site suppression model suggests that mutations tend to be deleterious to function, but that an amino acid change at a second site could suppress the mutation and restore

function.

The covarion model proposed by Fitch and Markowitz [30] was developed while they were investigating seemingly contradictory evidence regarding the neutral theory of molecular evolution [48]. They noted that only 10% of sites in cytochrome c could accept a mutation, which seemed to contradict the neutral theory that says that most mutations become fixed because they are selectively neutral to the function of the protein. However, the contradiction was resolved by the explanation that as mutations occur in the protein, the set of positions that can accept a mutation changes. New opportunities for change arise and sites that could previously accept change become immutable in the new sequence environment.

More recently, Weinreich et al. investigated five β-lactamase point mutations that significantly increased bacterial antibiotic resistance [81]. Of the 120 possible permutations, they determined that 102 were inaccessible to selection because they are deleterious. Further, many of the remaining pathways to the 5 necessary mutations are unlikely to occur because most do not convey any appreciable resistance. These data suggest that evolution is constrained to a specific narrow pathway on which it can act. The same group investigated compensatory mutations in dipteran genomes. Kulathinal et al. describe how residues which cause mutant phenotypes in *Drosophila melanogaster* produce wild-type phenotypes at orthologous sites in other dipteran genomes, suggesting that second site suppression has occurred [51]. However, they also acknowledge that these compensatory mutations, which occur under high selection in *D. melanogaster*, may be selectively neutral under other conditions.

Around the same time, Poon et al. measured the rate of compensatory mutation for 21 randomly-chosen, deleterious point mutations in DNA Bacteriophage ΦX174. They found that compensatory mutations occurred at a second site (rather than a reversion) in 70% of cases. They also highlighted intra-protein compensatory mutations that occurred in structurally-local positions [66]. In an analysis of the same dataset, Davis et al. demonstrated that compensatory mutations tended to occur near the deleterious mutation in sequence space [18].

An important observation is that both the second site suppression and covarion models im-

ply that sequence changes affect the positions nearby in three-dimensional space. Yanofsky et al. postulated that a 36 amino acid segment separating the mutation from the suppressor facilitated the contact between the two positions [82]. Fitch and Markowitz suggested that the mutations showed "spatial correlation" was related to selectively neutral changes; for example, two positions could accept a change in a selectively neutral manner provided that steric hindrance was avoided [30]. Both hypotheses imply coevolution is likely to occur between contacting positions in order to generate the compensatory change or alter the context surrounding the positions in question.

When analyzing a multiple sequence alignment, the first characteristic typically examined is conservation because a conserved amino acid that it is important to the enzyme's ability function [69]. Non-independent positions are clearly also important, but rather than imply an important position, correlated mutations imply an important, conserved *interaction* [34]. We call this type of positional non-independence coevolution.

1.4.2 Coevolution

Coevolution occurs within proteins when two orthologous positions influence one another's ability to change over evolutionary time [30]. It is similar to the coevolution that is seen between species, but it is amino acids influencing one another rather than organisms. At a coarse level, coevolution between positions within a protein can be detected by a method that identifies correlations between the positions. Essentially, one needs a method that takes a multiple sequence alignment as input and outputs pairwise covariation scores, from which one infers potentially coevolving positions. Unfortunately, correlation between positions within a protein family is not the only source of covariation between positions in an alignment.

It is important to distinguish between covariation and coevolution as the terms are sometimes used interchangeably to the detriment of the comprehension of the reader. Coevolution is the biological process through which two positions are actually influencing one another through evolutionary time. Covariation is the statistical method used to detect correlation between posi-

tions within a protein. Covariation can imply coevolution, but there are other potential causes. Atchley et al. suggested that the covariation signal in a protein could be broken down into many components including phylogenetic, stochastic, structure, function, and interactions [8]. It is possible that even more components exist. For the purposes of identifying coevolution, the stochastic and phylogenetic components are noise.

$$C_{ij} = C_{phylogeny} + C_{structure} + C_{function} + C_{interactions} + C_{stochastic}$$

1.4.3 Coevolution inference statistics

Many strategies exist for inferring intra-protein coevolution. The OMES method compares the observed frequency of amino acid pairs with the expected frequency based on the positional frequencies [52, 44]. OMES is a simple method that assumes that non-coevolving positions will have pair frequencies match the expected pair frequencies based on each respective positional frequency. This method is often a baseline tool that other tools are benchmarked against.

The McLachlin-based Substitution Correlation (McBASC) involves calculating a substitution matrix for each position[35]. The matrices can then be compared using a linear correlation to determine the extent to which each position covaries and the predicted likelihood that the positions are in contact. While seemingly more sophisticated, practical evaluations show that it performs approximately as well as OMES.

Statistical Coupling Analysis (SCA) is a perturbation-based covariance method proposed by Lockless and Ranganathan [56]. Perturbation-based covariance methods are based on subsampling the multiple sequence alignment and comparing results from the subsample with results from the entire alignment. SCA produces statistical coupling energy, a measure of coevolutionary dependence. Dekker et al. noted that the "metaphorical energy" values were not easily interpreted and thus, produced another perturbation-based covariance method, Explicit Likelihood of Subset Covariation (ELSC). ELSC improved contact prediction accuracy over SCA [21].

CAPS, Coevolution Analysis of Protein Sequences [27], is a parametric coevolution method,

meaning that it is based on a phylogenetic tree. The principal innovation of CAPS is to remove specific clades from the phylogenetic tree and recompute the coevolution analysis. If a pair is no longer detected, they classified it as being due to phylogenetic covariation. There was a division in the field as to whether phylogenetic tree information should be included in a coevolution-inference. While it is additional information to add to the model, it is also an additional source of error since phylogenetic trees of large numbers of sequences are necessarily approximations and hypotheses. In practical benchmarks, tree-based methods did not outperform tree-agnostic methods and thus I did not decide to include them in my own work.

Unlike pairwise methods, Direct-Coupling Analysis (DCA) involves calculating a global statistical model [61]. While a global statistical model may seem more likely to produce accurate results than a pairwise analysis, many pairwise methods apply corrections that incorporate an alignment-wide analysis. Although DCA is a recent method and is considered by some to be among the most accurate, the tool was benchmarked against a modified version of uncorrected MI that included gap characters as the 21st amino acid. I will discuss why this results in poor quality predictions in chapter 4. Thus it is difficult to truly ascertain the accuracy of DCA. However, benchmarks results do not suggest that DCA significantly outperforms *Zpx*, described in chapter 5.

Finally, there is a class of coevolution-inference tools that are based on the concept of Information Theory. Specifically, Mutual Information (*MI*) can be used to infer the dependence of two random variables, and thus can be used with various corrections to infer coevolution.

1.4.4 Mutual Information-based covariation

Mutual information is an Information Theoretic quantity [7, 15] on which many coevolution-inferring statistics have been based:

$MI_{i,j} = H_i + H_j - H_{i,j}$

MI is a measure of the dependence of one random variable upon another and is dependent on Information Entropy: *H*. Entropy is the measure of uncertainty of a random variable, in this

case the probability (P) that an amino acid (x) appears at a position in the alignment.

$$H_i = -\sum P(x_i) log_{20} P(x_i)$$

One of the most obvious problems in analyzing sequence alignment positions in a pairwise manner is the availability of data, since there is often insufficient sequence variety available within a typical orthologous protein family to adequately identify the "true" probability of an amino acid occurring at a given position. A "complete" multiple sequence alignment for a given protein family would consist of every extant orthologous protein, at minimum. Such a dataset would likely also need to contain all ancestral sequences of the extant dataset, and all potential descendent sequences. Such a dataset is obviously impossible to collect. Practicality requires an approximate solution which is found by using the frequency that an amino acid is observed at a position in place of the probability of it occurring. A pairwise frequency table for two potentially covarying positions contains 400 entries if all residues are equally likely. If the two pairs are, in reality, equally and identically assorting, one would need many times that number of sequences to establish the random assorting with confidence. Some estimates placed the value needed to assess covariation confidently at greater than ten thousand non-unique sequences; however, empirical observation shows that a practical minimum is approximately 125 sequences since the generation of reasonable contact prediction accuracy is possible with that number of sequences [60].

One of the major limitations of *MI* is that Mutual Information scores form a continuous distribution and it is difficult to determine where a cutoff should be drawn to differentiate the putatively coevolving pairs from those that are not coevolving. One strategy often used in benchmarking is to ignore the absolute *MI* score, and instead rank all pairs by *MI* score and select some fraction of them as the putatively-coevolving pairs. Often the fraction is defined by the length of the protein, under the assumption that longer proteins are more likely to have more coevolving pairs. This method is acceptable for benchmarking because the concern is the relative prediction accuracy of various methods. However, this method is not appropriate for inferring truly coevolving pairs because there is no way to know *a priori* how many coevolving

pairs exist within a protein family or whether it is some function of the length of the protein family.

Korber et al. used uncorrected *MI* to look for correlated mutations in a short, variable loop region of an HIV envelope protein and found a small set of correlated pairs of positions [50]. This work represents important first steps in the application of *MI* to coevolution-inference. This early work does not make reference to inter-residue contact and three-dimensional structure for validation. Instead, it attempts to link the putatively-coevolving residues with functional importance. The authors comment on the generalizability to other protein families.

Clarke analyzed residues in the homeodomain protein family and created a correction to account for the number of amino acid pairs between positions [13]. This work established the importance of inter-residue contact as it features a number of strongly-covarying salt bridges. However, this modified Mutual Information-based statistic yielded unexplained and likely false-positive results as well. Clarke speculates that the results could be the result of either statistical artifacts, or truly coevolving positions whose structural or functional role had yet to be observed.

Atchley et al. used *MI* to analyze basic helix-loop-helix (bHLH) protein family [8]. They asserted the importance of low entropy (ie. conserved) positions to the structure of the then-242 sequence bHLH protein family. Furthermore, they used a parametric bootstrap procedure to attempt to infer the probability that given covarying pair was covarying due to various components of the linear model. For example, very strongly coevolving pairs had a low probability of covarying predominantly due to the phylogeny component.

Tillier et al. created a correction for *MI* to compensate for the phylogenetic background component of covariation by only selecting pairs that covaried strongly with one another, but not any other groups of positions[78]. This correction was effective at removing some some pairs that covaried due to shared phylogenetic signal; however, some true positive coevolving pairs may have been removed by this correction, because it is possible that networks of truly coevolving residues could form within a protein [30].

Fodor and Aldrich compared *MI* (without corrections) to other established methods in 2004 and determined that *MI* predicted residue contact poorly. *MI* was slightly worse than SCA and much worse than both McBASC and OMES at predicting inter-residue contact in PFAM alignments [31]. They observed that there was little agreement between predictions of the four major coevolution-inference statistics, and proposed that this was because each method was "filtering" the positions it analyzed based on conservation (ie. entropy).

Independently, a group at the University of Western Ontario observed that MI scores were very dependent on the entropy of the positions. This is not surprising because positions that show more variation are more able to show covariation; positions that do not vary cannot covary, by definition. This is an unacceptable bias for protein alignment data, because many positions show conservation because of structural of functional constraints. A bias towards high entropy positions obscures important coevolving pairs and a lower contact prediction accuracy [60]. They employed a normalization for entropy that decreased the dependence on entropy and increased protein structure prediction accuracy. Related work by Gloor et al. yielded a dataset of high quality multiple sequence alignments [34] that were made in the structure-based alignment tool Cn3D [40], the same tool used to create the Conserved Domain Database [59]. This is an important development because the alignments found in public databases were not of sufficient quality to yield reliable predictions with entropy-normalized MI.

Another limitation of uncorrected MI is its inability to account for the fact that all positions within a protein covary to some degree because of their shared phylogenetic ancestry [8]. If homologous, every single extant sequence in a family has evolved from a common ancestor. If each sequence had been evolved independently, then there would be no background phylogenetic component. This background phylogenetic covariation can obscure true coevolution signal.

Dunn et al., also at the University of Western Ontario, provided a more sophisticated correction that superseded the entropy normalization by removing the phylogenetic covariation component [22]. The new statistic, *MIp*, showed a significant increase in contact prediction ac-

curacy. Dunn et al. concatenated multiple sequence alignments of phylogenetically linked but non-interacting proteins to empirically observe the background phylogenetic signal between them. They approximated this phylogenetic signal in a correction called the *APC* that was subtracted from the *MI* score. *MIp* appeared to approximate a normal distribution. Z-scores are used to compare the significance of *MIp* scores between protein families. *MIp* Z-scores are referred to as *Zp*. An outlier in the normal distribution implies that pair violates the assumption that all pairs of positions are randomly assorting and thus are likely coevolving.

Gloor et al. further explored the phylogenetic placement of coevolving positions by analyzing the distribution of putatively-coevolving positions on the phylogenetic tree, finding that the most strongly covarying positions were not characterized by either strong synapomorphy or homoplasy [33], an interesting result because it had been debated whether coevolution should be characterized by early fixation with little change due to an evolutionary barrier, or frequent compensatory mutation with many derived synapomorphies defining many clades.

In conclusion, it is worth highlighting the critical importance of the concept of conservation to all evolution, but specifically sequence analysis. Conservation of amino acid identity at given positions is what is used to infer homology, create alignments, infer phylogeny, and infer the importance of special positions within a protein. When a position shows perfect conservation, it is automatically assumed that such a position is important to function lest it was changed by the random meanderings of evolution [69]. Coevolution can provide similarly important information about a protein and its positions. Where conservation implies an important position within the protein, coevolution implies an important *interaction* between positions. And coevolution has the potential to play a part in homology inference, alignment creation, phylogenetic inference, and highlighting residues critical for function. Coevolution has the potential to be as important to the study of proteins as conservation.

1.5 Summary

The overarching hypothesis for my thesis was that Mutual Information-based coevolution statistics depend upon the underlying dataset from which they are generated. Coevolution statistics contain explicit and implicit assumptions about the nature of the input alignment that must be understood and respected by the end-user. However, some have attempted to apply coevolution statistics and trust the results without acknowledging such assumptions and requirements. For example, an early assumption about the coevolution-alignment relationship was that coevolution predictions were an indicator of alignment quality and that covariation scores should be maximized when creating a multiple sequence alignment. My goal was to establish the link between alignment quality and MutualInformation-based coevolution predictions. I have the following objectives outlined in the following chapters:

In chapter 2, I describe in detail the process for reliably creating coevolution predictions using publicly-available tools, including LoCo, and the MIpToolset, both of which I wrote. The process involves collecting sequences using local alignment, creating a protein family alignment using structure-based global alignment, curating the alignment using LoCo to either remove or realign sequences, and how to analyze the protein family using the MIpToolset, including the production of a network representation of coevolving positions. It provides guidance and frequent pitfalls for researchers who wish to gain new insights into their protein family of interest.

In chapter 3, I discuss the algorithmic optimizations made to the MIpToolset for rapid coevolution predictions. One of the major challenges of coevolution prediction methods is the time and memory required to generate predictions for all pairs within a protein sequence. The matrix-population algorithm in the software adjoining Dunn et al. made database-wide analyses and analysis of large alignments slow [22]. The data structure-based solution described in chapter 3 allows for rapid analysis and facilitates the real-time editing calculations used in the LoCo tool described in chapter 6. Thus, chapter 3 is a description of a software package I wrote and have made available to the molecular coevolution research community.

In chapters 4 through 6, I address the limitations of *MIp* and the advances that I made to coevolutionary inference. Although *MIp* was a major advance, there were a number of major limitations. *MIp* appeared to be very dependent on the alignment: It performed well on the hand-curated dataset created for benchmarking coevolution statistics, but was inconsistent when analyzing alignments in public databases. In order for *MIp* and all of coevolutionary inference to be beneficial as a general tool, an explanation was needed for why *MIp* would produce high-quality predictions for some alignments and not for others. Comparing *MIp* predictions to the target structure was an excellent way to benchmark *MIp* predictions, but is of little benefit in *a priori* protein folding.

Large-scale analysis of public databases made possible by the development of the MIp-Toolset could then lead to an investigation of the reason for the inconsistency of *MIp*'s performance with the potential for either guidelines for choosing and building alignments for use with *MIp* or the formulation of a new version of *MIp* which was more robust to the heretofore unknown issues in public gold-standard databases like CDD.

In chapter 4, I investigate the relationship between the treatment of gaps in sequence alignments and the predictions of *MIp*. A commonly-held assumption in the field is that gaps are an acceptable source of coevolutionary information, and thus many groups include gaps as the "21st amino acid" in their coevolution inference method despite the fact that the original *MIp* description explicitly avoided positions with gaps. Because the assumption about gaps was untested, I investigated the hypothesis that gaps and gapped positions contain coevolutionary information and would yield meaningful contact predictions. I rejected this hypothesis; positions containing gaps do not produce high-quality contact predictions. I explain in detail the relationship between *MIp*, the alignment, and how gaps interrupt the analysis and create false-positive results.

In chapter 5, I continue the investigation of the relationship between the multiple sequence alignment and the quality of coevolution predictions. I tested the hypothesis that alignment quality in the form of presence of alignment errors affected the ability of *MIp* to find predict

inter-residue contacts. This was supported, though the results were contrary to my assumption that shift errors would cause a reduction in *MIp* score, effectively obscuring the coevolutionary signal. The opposite was true, as systematic alignment errors found in the Conserved Domain Database greatly increased the *MIp* score, resulting in a false-positive result. I then investigated whether it was possible to correct the misalignment would and restore the contact prediction ability of *MIp* on that alignment. Finally, I created a new normalization for Mutual Information that provides *MIp*-like contact prediction accuracy on public database alignments that could contain errors. The new covariation statistics Zpx and ΔZp provide greater contact prediction accuracy than *MIp* even in alignments predicted likely to contain systematic misalignments.

In chapter 6, I continue investigate alignment errors and how they relate to coevolution predictions. I hypothesized that the false-positive *MIp* pattern described in chapter 5, which I call local covariation, was repeatable in other multiple sequence alignments such that I could detect alignment errors using patterns in *MIp*. A corollary to this hypothesis is that correcting the misalignment will make the local covariation pattern go away, and thus I also investigated whether *MIp* could be used as a guide to find the correct alternative alignment. I analyzed the benchmark alignment database BAliBASE 3, and found many instances of high local covariation. When I investigated specific alignments for which there was structural evidence to determine the correct alignment, I found that the alternative alignment suggested by local covariation was supported by the structure, and that the alignment in BAliBASE 3 contradicted the structure. Chapter 6 also includes a description of my software tool LoCo, which is freely available online. LoCo provides real-time calculation and display of sequence-local *MIp*, called "local covariation", to identified potentially misaligned segments in a sequence alignment in a user-friendly graphical interface.

Bibliography

[1] Paul D Adams, David Baker, Axel T Brunger, Rhiju Das, Frank Dimaio, Randy J Read, David C Richardson, Jane S Richardson, and Thomas C Terwilliger. Advances, interactions, and future developments in the CNS, Phenix, and Rosetta structural biology software systems. *Annual review of biophysics*, 42:265–287, 2013.

[2] S F Altschul. Generalized affine gap costs for protein sequence alignment. *Proteins: Structure, Function, and Bioinformatics*, 32(1):88–96, July 1998.

[3] S F Altschul, W Gish, W Miller, E W Myers, and D J Lipman. Basic local alignment search tool. *Journal of Molecular Biology*, 215(3):403–410, October 1990.

[4] S F Altschul and E V Koonin. Iterated profile searches with PSI-BLAST–a tool for discovery in protein databases. *Trends Biochem Sci*, 23(11):444–447, November 1998.

[5] S F Altschul, T L Madden, A A Schaffer, J Zhang, Z Zhang, W Miller, and D J Lipman. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Research*, 25(17):3389–3402, September 1997.

[6] C B Anfinsen. Principles that govern the folding of protein chains. *Science*, 181(4096):223–230, July 1973.

[7] Robert B Ash. *Information Theory*. Courier Dover Publications, 1965.

[8] W R Atchley, K R Wollenberg, W M Fitch, W Terhalle, and A W Dress. Correlations among amino acid sites in bHLH protein domains: an information theoretic analysis. *Mol Biol Evol*, 17(1):164–178, January 2000.

[9] A Bahr, JD Thompson, JC Thierry, and O Poch. BAliBASE (Benchmark Alignment dataBASE): enhancements for repeats, transmembrane sequences and circular permutations. *Nucleic Acids Research*, 29(1):323, 2001.

[10] J Castresana. Selection of conserved blocks from multiple alignments for their use in phylogenetic analysis. *Mol Biol Evol*, 17(4):540–552, April 2000.

[11] H Cheng, B-H Kim, and N V Grishin. MALISAM: a database of structurally analogous motifs in proteins. *Nucleic Acids Research*, 36(Database):D211–D217, December 2007.

[12] FD Ciccarelli, T Doerks, C Von Mering, CJ Creevey, B Snel, and P Bork. Toward automatic reconstruction of a highly resolved tree of life. *Science*, 311(5765):1283, 2006.

[13] N D Clarke. Covariation of residues in the homeodomain sequence family. *Protein Sci*, 4(11):2269–2278, November 1995.

[14] Cn3d Tutorial. Structure Alignments in Cn3D. September 2011.

[15] T M Cover and Joy A Thomas. *Elements of information theory*. New York, 1991.

[16] F H Crick, L Barnett, S Brenner, and R J Watts-Tobin. General nature of the genetic code for proteins. *Nature*, 192:1227–1232, December 1961.

[17] Charles Darwin. *The Origin of Species*. London, 1859.

[18] B H Davis, A F Y Poon, and M C Whitlock. Compensatory mutations are repeatable and clustered within proteins. *Proceedings of the Royal Society B: Biological Sciences*, 276(1663):1823–1827, April 2009.

[19] Margaret O Dayhoff and R V Eck. *Atlas of protein sequence and structure* . National Biomedical Research Foundation., 1968.

[20] David de Juan, Florencio Pazos, and Alfonso Valencia. Emerging methods in protein co-evolution. *Nat Rev Genet*, 14(4):249–261, April 2013.

[21] John P Dekker, Anthony Fodor, Richard W Aldrich, and Gary Yellen. A perturbation-based method for calculating explicit likelihood of evolutionary co-variance in multiple sequence alignments. *Bioinformatics*, 20(10):1565–1572, July 2004.

[22] S D Dunn, L M Wahl, and G B Gloor. Mutual information without the influence of phylogeny or entropy dramatically improves residue contact prediction. *Bioinformatics*, 24(3):333–340, January 2008.

[23] Sean R Eddy. Accelerated Profile HMM Searches. *PLoS Computational Biology*, 7(10):e1002195, October 2011.

[24] RC Edgar. Quality measures for protein alignment benchmarks. *Nucleic Acids Research*, 38(7):2145, 2010.

[25] Robert C Edgar. MUSCLE: a multiple sequence alignment method with reduced time and space complexity. *BMC Bioinformatics*, 5:113, August 2004.

[26] Gang Fang, Nitin Bhardwaj, Rebecca Robilotto, and Mark B Gerstein. Getting Started in Gene Orthology and Functional Analysis. *PLoS Computational Biology*, 6(3):e1000703, March 2010.

[27] Mario A Fares and David McNally. CAPS: coevolution analysis using protein sequences. *Bioinformatics*, 22(22):2821–2822, November 2006.

[28] Robert D Finn, Jody Clements, and Sean R Eddy. HMMER web server: interactive sequence similarity searching. *Nucleic Acids Res*, 39(Web Server issue):W29–37, July 2011.

[29] W M Fitch. Distinguishing homologous from analogous proteins. *Systematic zoology*, 19(2):99–113, June 1970.

[30] W M Fitch and E Markowitz. An improved method for determining codon variability in a gene and its application to the rate of fixation of mutations in evolution. *Biochem Genet*, 4(5):579–593, 1970.

[31] Anthony A Fodor and Richard W Aldrich. Influence of conservation on calculations of amino acid covariance in multiple sequence alignments. *Proteins: Structure, Function, and Bioinformatics*, 56(2):211–221, 2004.

[32] J F Gibrat, T Madej, and S H Bryant. Surprising similarities in structure comparison. *Curr Opin Struct Biol*, 6(3):377–385, June 1996.

[33] GB Gloor, G Tyagi, DM Abrassart, AJ Kingston, AD Fernandes, SD Dunn, and CJ Brandl. Functionally Compensating Coevolving Positions Are Neither Homoplasic Nor Conserved in Clades. *Molecular Biology and Evolution*, 27(5):1181, 2010.

[34] Gregory B Gloor, Louise C Martin, Lindi M Wahl, and Stanley D Dunn. Mutual information in protein multiple sequence alignments reveals two classes of coevolving positions. *Biochemistry*, 44(19):7156–7165, May 2005.

[35] U Gobel, C Sander, R Schneider, and A Valencia. Correlated mutations and residue contacts in proteins. *Proteins: Structure, Function, and Bioinformatics*, 18(4):309–317, 1994.

[36] A Godzik. The structural alignment between two proteins: is there a unique answer? *Protein Sci*, 5(7):1325–1338, July 1996.

[37] E M Golenberg, M T Clegg, M L Durbin, J Doebley, and D P Ma. Evolution of a noncoding region of the chloroplast genome. *Molecular phylogenetics and evolution*, 2(1):52–64, March 1993.

[38] Barry G Hall. Simple and accurate estimation of ancestral protein sequences. *Proceedings of the National Academy of Sciences of the United States of America*, 103(14):5431–5436, April 2006.

[39] 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(22):10915–10919, November 1992.

[40] C W Hogue. Cn3D: a new generation of three-dimensional molecular structure viewer. *Trends Biochem Sci*, 22(8):314–316, August 1997.

[41] T H H Huxley. The Origin of Species. *Westminister Review*, 17:541–570, 1860.

[42] Kristoffer Illergård, David H Ardell, and Arne Elofsson. Structure is three to ten times more conserved than sequence–a study of structural response in protein cores. *Proteins: Structure, Function, and Bioinformatics*, 77(3):499–508, November 2009.

[43] L.A. Johnson and D.E. Soltis. Phylogenetic inference in Saxifragaceae sensu stricto and Gilia (Polemoniaceae) using matK sequences. *Annals of the Missouri Botanical Garden*, pages 149–175, 1995.

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

[45] Kazutaka Katoh and Hiroyuki Toh. Recent developments in the MAFFT multiple sequence alignment program. *Brief Bioinform*, 9(4):286–298, July 2008.

[46] Kazuharu Misawa Kei-ichi Kuma Takashi Miyata Kazutaka Katoh. MAFFT: a novel method for rapid multiple sequence alignment based on fast Fourier transform. *Nucleic Acids Research*, 30(14):3059, July 2002.

[47] Changhoon Kim and Byungkook Lee. Accuracy of structure-based sequence alignment of automatic methods. *BMC Bioinformatics*, 8:355, 2007.

[48] M Kimura. Evolutionary rate at the molecular level. *Nature*, 217(5129):624–626, 1968.

[49] Eugene V Koonin. Orthologs, paralogs, and evolutionary genomics. *Annual review of genetics*, 39:309–338, 2005.

[50] B T Korber, R M Farber, D H Wolpert, and A S Lapedes. Covariation of mutations in the V3 loop of human immunodeficiency virus type 1 envelope protein: an information theoretic analysis. *Proc Natl Acad Sci U S A*, 90(15):7176–7180, 1993.

[51] R J Kulathinal. Compensated Deleterious Mutations in Insect Genomes. *Science*, 306(5701):1553–1554, November 2004.

[52] S M Larson, A A Di Nardo, and A R Davidson. Analysis of covariation in an SH3 domain sequence alignment: applications in tertiary contact prediction and the design of compensating hydrophobic core substitutions. *J Mol Biol*, 303(3):433–446, October 2000.

[53] Ying Liu and Ivet Bahar. Sequence Evolution Correlates with Structural Dynamics. *Molecular Biology and Evolution*, April 2012.

[54] Ying Liu, Eran Eyal, and Ivet Bahar. Analysis of correlated mutations in HIV-1 protease using spectral clustering. *Bioinformatics*, 24(10):1243–1250, May 2008.

[55] Ying Liu, Lila M Gierasch, and Ivet Bahar. Role of Hsp70 ATPase domain intrinsic dynamics and sequence evolution in enabling its functional interactions with NEFs. *PLoS Computational Biology*, 6(9), 2010.

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

[57] A Loytynoja and N Goldman. Phylogeny-Aware Gap Placement Prevents Errors in Sequence Alignment and Evolutionary Analysis. *Science*, 320(5883):1632–1635, June 2008.

[58] Ari Löytynoja and Nick Goldman. webPRANK: a phylogeny-aware multiple sequence aligner with interactive alignment browser. *BMC Bioinformatics*, 11:579, 2010.

[59] Aron Marchler-Bauer, John B Anderson, Farideh Chitsaz, Myra K Derbyshire, Carol DeWeese-Scott, Jessica H Fong, Lewis Y Geer, Renata C Geer, Noreen R Gonzales, Marc Gwadz, Siqian He, David I Hurwitz, John D Jackson, Zhaoxi Ke, Christopher J Lanczycki, Cynthia A Liebert, Chunlei Liu, Fu Lu, Shennan Lu, Gabriele H Marchler, Mikhail Mullokandov, James S Song, Asba Tasneem, Narmada Thanki, Roxanne A Yamashita, Dachuan Zhang, Naigong Zhang, and Stephen H Bryant. CDD: specific functional annotation with the Conserved Domain Database. *Nucleic Acids Res*, 37(Database issue):D205–10, 2009.

[60] L C Martin, G B Gloor, S D Dunn, and L M Wahl. Using information theory to search for co-evolving residues in proteins. *Bioinformatics*, 21(22):4116–4124, November 2005.

[61] Faruck Morcos, Andrea Pagnani, Bryan Lunt, Arianna Bertolino, Debora S Marks, Chris Sander, Riccardo Zecchina, José N Onuchic, Terence Hwa, and Martin Weigt. Direct-coupling analysis of residue coevolution captures native contacts across many protein families. *Proceedings of the National Academy of Sciences*, 108(49):E1293–301, December 2011.

[62] J Moult, J T Pedersen, R Judson, and K Fidelis. A large-scale experiment to assess protein structure prediction methods. *Proteins: Structure, Function, and Bioinformatics*, 23(3):ii–v, November 1995.

[63] S B Needleman and C D Wunsch. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J Mol Biol*, 48(3):443–453, March 1970.

[64] C Notredame, D G Higgins, and J Heringa. T-Coffee: A novel method for fast and accurate multiple sequence alignment. *J Mol Biol*, 302(1):205–217, September 2000.

[65] L Pauling and E Zuckerkandl. Chemical Paleogenetics. *ACTA CHEMICA SCANDINAVICA*, 17:9–16, 1963.

[66] Art Poon and Lin Chao. The rate of compensatory mutation in the DNA bacteriophage phiX174. *Genetics*, 170(3):989–999, 2005.

[67] G P S Raghava, Stephen M J Searle, Patrick C Audley, Jonathan D Barber, and Geoffrey J Barton. OXBench: a benchmark for evaluation of protein multiple sequence alignment accuracy. *BMC Bioinformatics*, 4:47, October 2003.

[68] RB Russell and GJ Barton. Multiple protein sequence alignment from tertiary structure comparison: assignment of global and residue confidence levels. *Proteins: Structure, Function, and Bioinformatics*, 14(2):309–323, 1992.

[69] Ora Schueler-Furman and David Baker. Conserved residue clustering and protein structure prediction. *Proteins: Structure, Function, and Bioinformatics*, 52(2):225–235, August 2003.

[70] Robert W Scotland. Deep homology: a view from systematics. *BioEssays : news and reviews in molecular, cellular and developmental biology*, 32(5):438–449, May 2010.

[71] M P Simmons and H Ochoterena. Gaps as characters in sequence-based phylogenetic analyses. *Systematic biology*, 49(2):369–381, June 2000.

[72] Janardanan Sreekumar, Cajo J F ter Braak, Roeland C H J van Ham, and Aalt D J van Dijk. Correlated mutations via regularized multinomial regression. *BMC Bioinformatics*, 12:444, 2011.

[73] G Talavera and J Castresana. Improvement of phylogenies after removing divergent and ambiguously aligned blocks from protein sequence alignments. *Systematic biology*, 56(4):564, 2007.

[74] W R Taylor. Residual colours: a proposal for aminochromography. *Protein Eng*, 10(7):743–746, July 1997.

[75] J D Thompson, D G Higgins, and T J Gibson. CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Res*, 22(22):4673–4680, November 1994.

[76] JD Thompson, F Plewniak, and O Poch. BAliBASE: a benchmark alignment database for the evaluation of multiple alignment programs. *Bioinformatics*, 15(1):87, 1999.

[77] Julie D Thompson, Patrice Koehl, Raymond Ripp, and Olivier Poch. BAliBASE 3.0: latest developments of the multiple sequence alignment benchmark. *Proteins: Structure, Function, and Bioinformatics*, 61(1):127–136, October 2005.

[78] ERM Tillier and TWH Lui. Using multiple interdependency to separate functional from phylogenetic correlations in protein alignments. *Bioinformatics*, 19(6):750–755, 2003.

[79] Ivo Van Walle, Ignace Lasters, and Lode Wyns. SABmark—a benchmark for sequence alignment that covers the entire known fold space. *Bioinformatics*, 21(7):1267–1268, 2005.

[80] Andrew M Waterhouse, James B Procter, David M A Martin, Michèle Clamp, and Geoffrey J Barton. Jalview Version 2–a multiple sequence alignment editor and analysis workbench. *Bioinformatics*, 25(9):1189–1191, May 2009.

[81] Daniel M Weinreich, Nigel F Delaney, Mark A DePristo, and Daniel L Hartl. Darwinian evolution can follow only very few mutational paths to fitter proteins. *Science*, 312(5770):111–114, 2006.

[82] C Yanofsky, V Horn, and D Thorpe. Protein structure relationships revealed by mutational analysis. *Science*, 146:1593–1594, 1964.

[83] C M Zmasek and S R Eddy. A simple algorithm to infer gene duplication and speciation events on a gene tree. *Bioinformatics*, 17(9):821–828, September 2001.

Chapter 2

Methods

Summary

Positions in a protein are thought to coevolve to maintain important structural and functional interactions over evolutionary time. The detection of putative coevolving positions can provide important new insights into a protein family in the same way that knowledge is gained by recognizing evolutionarily conserved characters and characteristics. Putatively coevolving positions can be detected with statistical methods that identify covarying positions. However, positions in protein alignments can covary for many other reasons than coevolution; thus, it is crucial to create high quality multiple sequence alignments for coevolution inference. Furthermore, it is important to understand common signs and sources of error. When confounding factors are accounted for, coevolution is a rich resource for protein engineering information.

A version of this manuscript has been accepted for publication.
RJ Dickson, GB Gloor. (2013). Bioinformatics Identification of Coevolving Residues. Methods in Molecular Biology. In press.

2.1 Introduction

Families of aligned proteins are a rich resource of structural, functional, and, evolutionary information. A standard intuition about homologous protein families is that evolutionarily conserved features structural or sequential positions and motifs can be seen as functionally important; if not, they would have changed through the random meanderings of evolutionary change [21, 22]. The critical point is that mutations may occur in a probabilistically random fashion, but the process by which they are maintained is non-random. Mutations that are detrimental to fitness are less likely to be retained. Thus conserved features are hypothesized to correspond with functional importance [23].

Intra-molecular coevolution coevolution within a protein sequence is potentially as robust a heuristic as conservation to determine functionally important positions. While conservation indicates catalytically critical residues (like Serine in a Serine Protease) or critical residue properties (like the acidic residues occupying the catalytic site of a LAGLIDADG homing endonuclease), it cannot easily indicate context dependent conservation or conserved interactions.

Coevolution analysis is a statistical method that identifies important structural and functional interactions from sequence. Coevolution is detected by identifying covariation between homologous sequence positions within a protein family. Coevolving positions have been shown to be much more likely to be in contact than chance; the coevolution-contact relationship is supported by the hypothesized mechanisms of protein evolution.

While coevolution tells us much about a protein, it depends entirely on the protein alignment from which it extracts information. Alignment errors resulting from both sequence collection and sequence shift result in false positive results in not only coevolution inference [8, 10], but also phylogenetic inference [32, 9]; many coevolution statistics cannot differentiate between covariation due to error or real evolutionary signal.

The sensitivity of coevolution statistics to error creates a necessity for protein family alignments of the highest quality. Errors in analysis can easily propagate and lead to false conclusions. There is a major tradeoff in sequence selection because the inclusion of homologous

proteins with non-identical function (eg. paralogues) can lead to false-positive identification of coevolving pairs [7]; conversely, it is crucial to maximize the number and diversity of the sequence alignment in order to produce accurate predictions [27]. Furthermore, the alignment of the putatively orthologous members of the protein family is equally crucial as alignment errors are known to produce false-positive results [8, 10].

Manual curation of an alignment, inspecting and revising an alignment by shifting and deleting sequences after the initial alignment procedure, is a crucial step to obtaining the highest quality coevolutionary signal. Several recent studies have shown that human curators outperform automated algorithmic solutions on protein structure and alignment problems [18, 20]. The drawback of manual curation of a protein alignment is the additional time needed and the potential for human error; in a large alignment it is easy to overlook alignment errors. The compromise solution between automation and accuracy is covariation-guided curation, where software identifies the potentially erroneous region, but a curator is required to make a ultimate decision on the final alignment. LoCo [9] uses a covariation-based heuristic, local covariation, to identify systematic misalignments within a modified Jalview [5, 33] interface for real-time alignment editing.

Herein we describe how to collect sequences using PSI-BLAST [24], build a structure-guided master-slave sequence alignment using Cn3D [17], refine that alignment using LoCo [9], calculate coevolutionary statistics using the MIpToolset [7], and finally visualize the network of coevolving pairs in contact map and network format. This analysis pipeline is thorough and complete and has lead to the identification of putative coevolving pairs in many protein families. However, the pipeline can also be viewed as somewhat modular, in that it is possible for an expert user to replace a given step of the analysis with a different methodology. For example, Hidden Markov Model-based homologue collection [30, 13] may provide a different set of homologous protein sequences than PSI-BLAST [2], and pre-computed alignments like those in CDD [26] and PFAM [31] can be used as time-saving resources; likewise, there are many different sequence alignment methods including MAFFT [19], or PRANK [25] which

could be used in place of MUSCLE [14] to align sequences when no structural information is available. Nonetheless, we must advise caution when making such substitutions in the analysis pipeline. Coevolution analysis is not robust to the many sources of error that routinely arise in homology detection and sequence alignment [11]. Thorough analysis in early steps will yield more reliable coevolution data.

2.2 Materials

Since this is a bioinformatics approach, the only physical requirement is a computer with a working Internet connection. All software and databases are freely available online. Some required software expects a Unix-like interface, so Mac OS X and Linux users should find the set-up straightforward; Windows users will have to emulate this by installing additional software for certain steps.

2.2.1 Computer and general-purpose software

1. Computer with an operating system that has a Unix-like interface. Common examples of Unix-like operating systems include Mac OS X and Linux. This solution has been tested on Mac OS X 10.5 through 10.8 and Ubuntu Linux. Windows users, see Note: 1.

2. GCC, the GNU compiler collection, is used to build some software outlined later (http://gcc.gnu.org/). The gcc compiler and make utility are necessary to build some platform-agnostic tools from source code. See Note: 2.

3. Perl: a programming language commonly used in bioinformatics applications (http://www.perl.org/get.html) and pre-installed on most Unix-like operating systems.

4. Java: a common portable programming language (http://www.java.com/en/).

2.2.2 Sequence collection tools and databases

1. Target sequence: the FASTA sequence of a protein you are studying or of a representative member of the family that you are studying. See Note: 3.

2. Target structure (optional): the PDB ID of the 3D structure of the target sequence or an orthologue of the target sequence. If a structure is available, we will build an alignment using section 2.3.3a. If a structure is not available, we will use sequence-only tools for building the alignment in 2.3.3b.

3. PSI-BLAST (Position-Specific Iterated Basic Local Alignment Search Tool) is a search tool designed to infer sequences that are homologous to a query sequence [24]

(ftp://ftp.ncbi.nih.gov/blast/executables/LATEST/). It is available as part of the BLAST+ package. Executables are available for many different operating systems.

4. There are many databases available for sequence collection. Suggested databases for this pipeline include nr and optionally nr_env (ftp://ftp.ncbi.nih.gov/blast/db/). See Note: 4.

5. readseq is a Java-based bioinformatics file format conversion tool 9,10,21. Download readseq.jar from (http://iubio.bio.indiana.edu/soft/molbio/readseq/java/).

2.2.3 Sequence alignment tools

1. Cn3D is a structure-guided sequence alignment tool

(http://www.ncbi.nlm.nih.gov/Structure/CN3D/cn3dinstall.shtml) used to build and curate the CDD database [26].

2. MUSCLE is an iterative sequence alignment tool (http://www.drive5.com/muscle/) [14] which seems to have parameters well-suited for creating protein alignments for coevolution analysis at the time of publication of this manuscript.

2.2.4 Alignment curation tools

1. LoCo [9] (http://sourceforge.net/projects/locoprotein/) is a modified version of the popular Jalview [5, 33] alignment tool. LoCo is used to curate (ie. validate and improve) protein alignments. See Note: 5.

2.2.5 Coevolution analysis

1. The MIpToolset [11, 7] (http://sourceforge.net/projects/miptoolset/) is a collection of C and Perl programs that calculates several popular Mutual Information-based coevolution metrics, *MI* [3, 6], *MIp/Zp* [12], *Zpx* [8], and ΔZp [11].

2. Graphviz is visualization software used to display networks of coevolving residues [15] (http://www.graphviz.org).

3. R is a statistical programming language which can be used to plot contact map and contact map figures [29] (http://www.r-project.org).

2.3 Methods

2.3.1 Building and installing bioinformatics tools

1. Download LoCo from (http://sourceforge.net/projects/locoprotein/) and the MIpToolset from (http://sourceforge.net/projects/miptoolset/). Unzip these programs into the desired install location (eg. your bin or Applications folder).

2. Build LoCo by following the directions in README_LoCo.txt, which is in the main LoCo directory. In brief, open a Terminal window on your computer. Change directory (*cd*) to the location where LoCo was unzipped. Then change directory to the

dist/MIpToolset_jalview/MIP_C_CODE/

directory and then type make to build the program.

3. The MIpToolset is built similarly; instructions are found in the README file and ad-

ditional information is found in the DOCUMENTATION folder. Inside the main MIpToolset directory is a directory called MIP_C_CODE. Change directory to the MIP_C_CODE subdirectory and run *make*, by typing *make* into the terminal. This will read the makefile and build two programs which calculate coevolution statistics and inter-residue distances, MIp and dist_pdb. An MIpToolset wrapper Perl script will access MIp, dist_pdb, and other Perl scripts when calculating covariation scores.

4. Additionally, you should make sure to add the installation directory to your *$PATH* environment variable by following the directions according to your operating system.

2.3.2 Collecting protein sequences and structures

1. Search for protein sequences that are homologous to your target protein by running the PSI-BLAST program installed in section 2.3.1.1. See Note: 6.

2. Store your target sequence in FASTA format in a file called "target.fa". See Note: 7.

3. Download the nr and nr_env databases and store them in an accessible directory like /data/. See Note: 4.

4. Select an initial Expectation value (*E value*) cutoff by using the following heuristic:

1E-X where *X* is (the length of the target sequence / 10). Round *X* to the nearest whole number. For example, if the protein is 200 residues long, choose *1E-20*. See Note: 7.

5. Run PSI-BLAST by executing the following from the terminal with several substitutions:

blastpgp -i target.fa -j 12 -d DATABASE_PATH -m 4 -e E_VALUE -I t -o psiblast.out

Substitute *DATABASE_PATH* with the location of the nr database and the name of the database from step 2.3.2.3; for example, if the database is stored in /data/ then replace *DATABASE_PATH* with */data/nr*. Replace *E_VALUE* with the value selected in 2.3.2.4; for example, *1E-10* if the protein length is 100.

Therefore, an example with substitutions is:

blastpgp -i target.fa -j 12 -d /data/nr -m 4 -e 1E-10 -I t -o psiblast.out

6. Ensure that the PSI-BLAST converged by inspecting the *psiblast.out* file created by the

blastpgp command.

7. Convert your blast output into FASTA format using readseq.jar. This program is a java archive (.jar) file and is run from the command line terminal using Java. In the following, it is assumed that the readseq.jar file is in the current directory; if it is not, simply replace readseq.jar with the relative or absolute path to the file. Type the following into the command line terminal:

java -jar readseq.jar -degap=- -f 8 psiblast.out

which should create the file psiblast.out.fa.

8. Inspect your alignment to determine whether you have enough sequences to continue. Ideally, you want at least 150 orthologous sequences with less than 90% sequence identity, so you should have approximately three hundred sequences in the dataset as a minimum to ensure sufficient sequences following the downstream filtering steps. Consider completing section 2.3.2 again using *nr_env* in place of *nr* to collect even more sequences.

2.3.3 Building a structure-guided sequence alignment (Alternately, use 2.3.4 for a sequence-only alignment if no structure is available.)

1. *Network Load* your target structure in Cn3D by selecting the *Network Load* option from the *File* menu and entering either the PDB or MMDB identifier. This will render the protein structure in the open *Structure Viewer* window and open the corresponding sequence in the *Alignment Viewer* window. See Note: 9.

2. Adjust the appearance of your target structure by selecting *Style* → *Rendering shortcuts* → *Tubes*, which aids in comparing structural alignment, and *Style* → *Coloring shortcuts* → *Sequence Conservation* → *Fit*, which colours the sequence and structure according to the position specific scoring matrix (PSSM) of the sequence alignment. See Note: 10.

3. Open the *Import Viewer* window by selecting *Imports* → *Show Imports*. Arrange your windows so that the *Structure, Alignment*, and *Import windows* are all visible. Import new structures into the *Import Viewer* window by selecting *Edit* → *Import Structure* and then *Via*

Network to enter the PDB or MMDB identifier, or *From a File* followed by a .pdb file (Figure 3.1). See Note: 11.

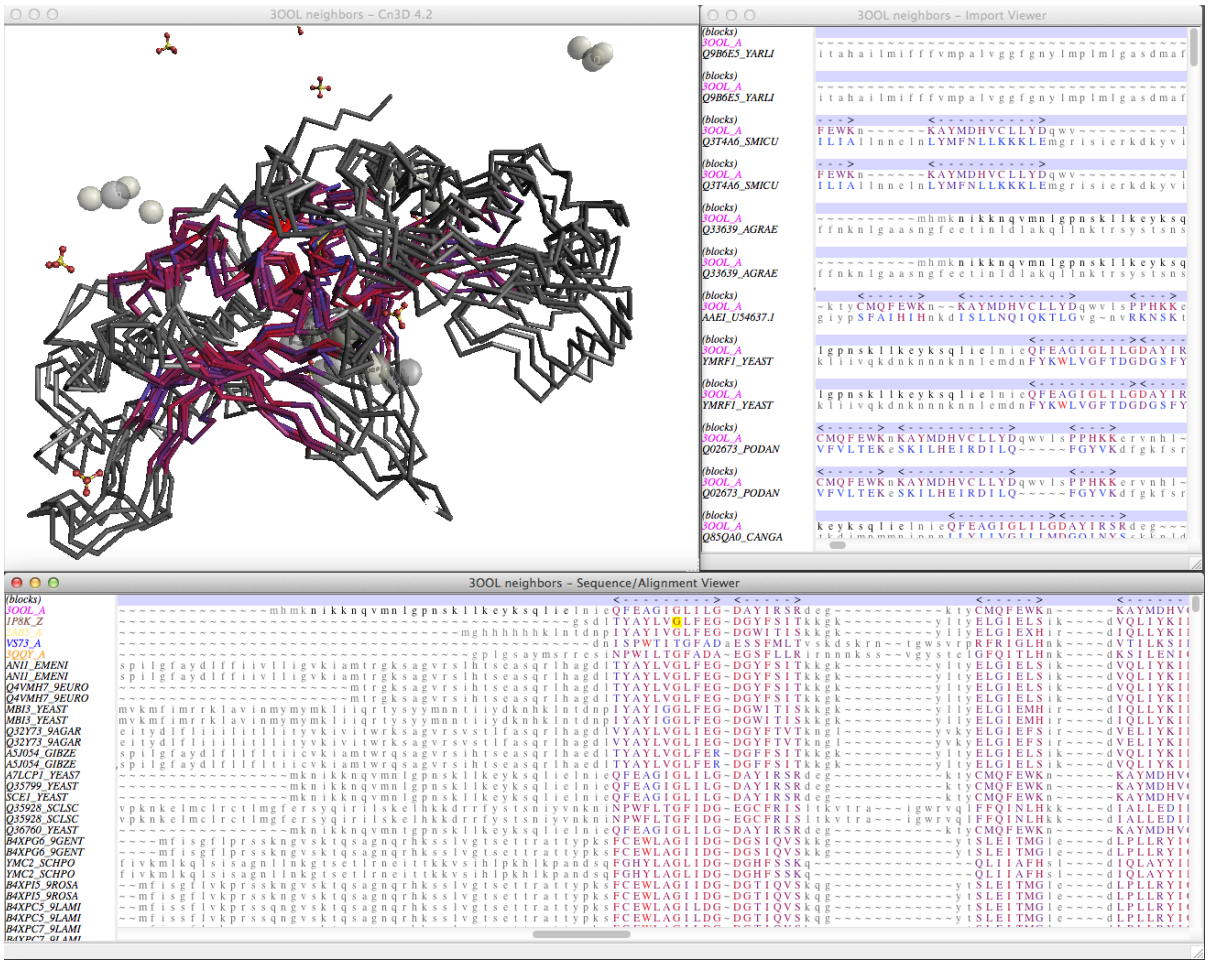

Figure 2.1: Screenshot of the three windows comprising the Cn3D workspace showing an analysis of the LAGLIDADG Homing Endonuclease family. The Structure Viewer window (top left) shows a protein structure alignment of the protein family. The Sequence Viewer window (bottom) shows the sequence alignment which corresponds to the structure alignment with additional sequences from the protein family. The Import Viewer window (rop right) shows sequences which have have been imported into Cn3D, but have not been added to the sequence alignment.

4. Attempt to *Merge All* the new structures from the *Import Viewer* into the *Alignment Viewer* by selecting Alignments → *Merge All*. There may be conflicts when your new structures are imported (residues highlighted in pink in the *Import Viewer*); conflicts will prevent structures from being added to the *Alignment Window* via the *Merge All* command.

5. Cn3D separates an alignment into block and gap sections, defined by the row labeled

blocks located above the alignments in the *Alignment Viewer*. View the blocks in the *Alignment Viewer* by selecting *Edit* → *Enable Editor*. Each block defines a structurally conserved segment of the alignment which cannot accept insertions or deletions. Adjust the each block in the *Alignment Viewer* so that the new structures in the *Import Viewer* no longer conflict; *Split, Merge, Create, Delete* (under the *Edit* menu), and *Horizontal Drag* (under the *Mouse Mode* menu) the blocks to resolve all pink conflicts in the *Import Viewer* window. See Note: 12.

6. Transfer the structures from the *Import Viewer* into the *Alignment Viewer* by selecting *Alignments* → *Merge All* in the *Import Viewer* window. Save your work to a file by selecting *Save* from the *File* menu in the *Structure Viewer* window. After saving, re-open the file to view the imported structures. Cn3D will not render the newly imported structures until the structures are merged into the Alignment Viewer and the file is saved and re-opened.

7. Select *File* → *Realign Structures* to superimpose the newly merged structures in the Structure Viewer as defined by the sequence relationships in the *Alignment Viewer*. Repeat this step whenever a change is made to the blocks or any structure in the *Alignment Window*, to update the structure alignment. See Note: 13.

8. Now that all structures are visible, revise the blocks to reflect the structural diversity of the protein family. Each block defines a conserved and critical structural feature that cannot accept insertions or deletions. Adjust the blocks in the Alignment Viewer so that they correspond to the structurally conserved core of the protein. Create gaps between the blocks in regions of structural divergence where insertions or deletions would be tolerated. As in step 5, use the options in the *Edit* and *Mouse Mode* menus to make changes. See Note: 14.

9. Structure alignment algorithms are vulnerable to shift errors, where major structural elements are superimposed, but individual residues are not (Figure 3.2); for example, two β-sheets may be superimposed, but in a configuration where aligned residues sidechains point in opposite directions. Search the alignment for shift errors by highlighting aligned residues (*Mouse Mode* → *Select Column* in the *Alignment Viewer*) and examining the corresponding structural alignment in the *Structure Viewer*; highlighted residues are highlighted in yellow in

both windows. Select alternate residues in the *Structure Viewer* by double clicking. Use both sequence homology and structural evidence to revise the structure alignment. Be thorough, as errors at this step will be propagated through the rest of the alignment creation process.

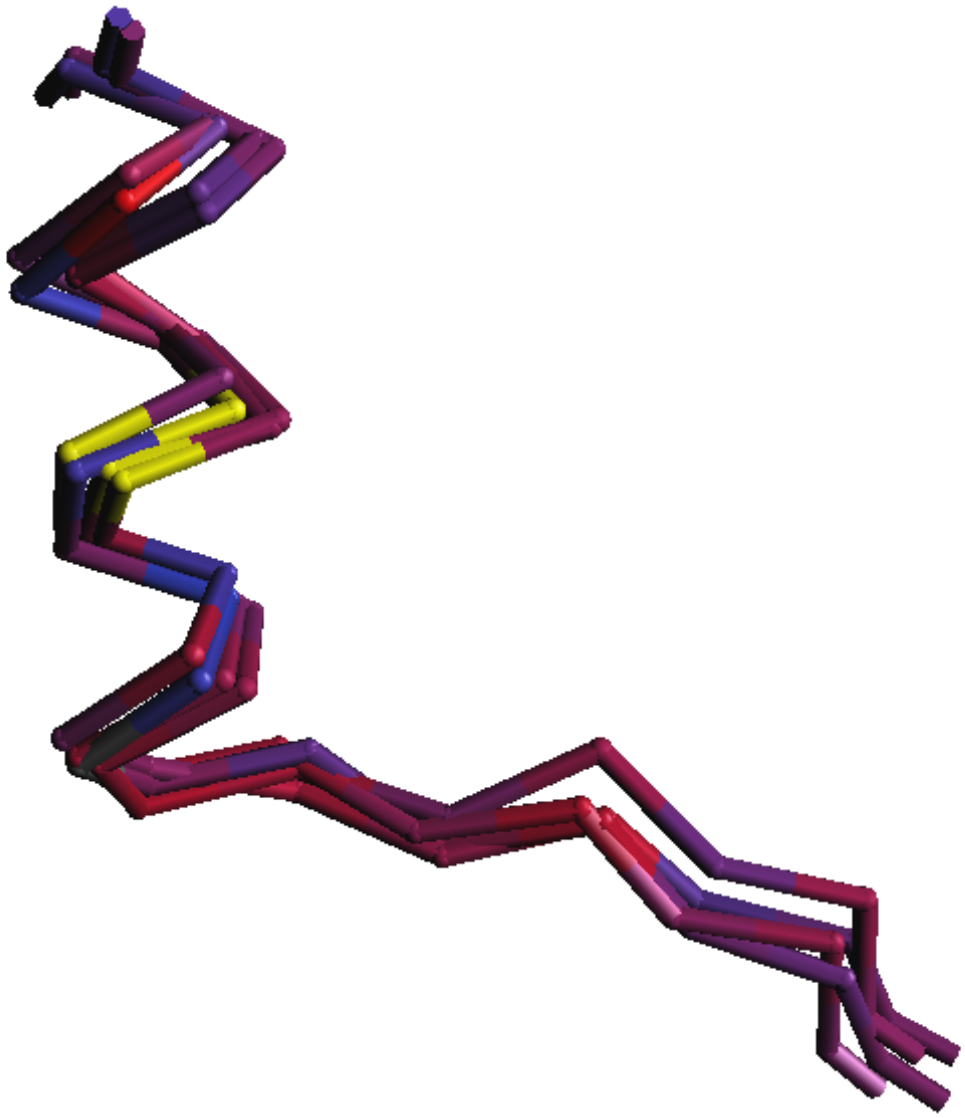

Figure 2.2: An example of a shift error added to a segment of a LAGLIDADG alignment. A vertical column in the sequence alignment was selected which highlights the corresponding residues in yellow in the Structure Viewer. One of the structures helices is out of alignment by 1 residue turn. This should be corrected by shifting the corresponding sequence in the sequence alignment.

10. Import the sequences collected by PSI-BLAST by selecting *Edit* → *Import Sequences* from the *Import Viewer* window; then select *From File* and finally the name of the PSI-BLAST

fasta file. These imported sequences will behave as the structures did in this window.

11. Align the new sequences to the target structure by selecting *Algorithms* → *Block Align N*; this procedure aligns sequences in the *Import Viewer* to the PSSM of the sequence alignment in the *Alignment Viewer*. After the block align procedure is complete, merge the sequences that fit with the existing block model. See Note: 15.

12. Sort the sequences by selecting *Edit* → *Sort Rows* → *By Score* and then *Edit* → *Sort Rows* → *Float PDBs* in the Alignment Viewer window. Inspect the alignment in the *Alignment Viewer*. The *Fit* colouring scheme will colour residues red that fit will with the PSSM for that position and blue for residues that fit poorly. Each sequence in the *Alignment Viewer* contributes to the PSSM, so poorly aligned sequences should be moved back to the *Import Viewer* by selecting *Imports* → *Realign Rows from List*.

13. In the *Import Viewer* perform a *Block Align N* to realign the sequences to to the expanded PSSM defined by the *Alignment Viewer*. Once again *Merge All* the sequences into the *Alignment Viewer*.

14. Repeat steps 12 and 13 iteratively until the alignment in the *Alignment Viewer* contains as many homologous sequences that will fit with both the block model and the sequence fit. See Note: 16.

2.3.4 Creating a sequence-only alignment (Alternately, use 2.3.3 for a structure-guided alignment.)

1. Run muscle on the collected sequences by opening a command line terminal and running:

muscle -in psi_blast.fasta -out alignment.fasta

where *psi_blast.fasta* is the file containing the FASTA-formatted sequences you collected in step 2.2.3.2. See Note: 17.

2.3.5 Curating and validating an alignment

1. Follow LoCo instructions to start the software; enter the LoCo directory and run the shell script by typing: . */run_loco.sh*

Open your alignment from the File menu by selecting Input Alignment - from file. See Note: 18.

2. Set the colour scheme from the Colour menu; it is good to start with a general-purpose colour scheme like Zappo. Zappo colours the sequence by amino acid properties so patterns can emerge visually. See Note: 19.

3. Sequences are manipulated in LoCo (and Jalview) as follows: Selected sequences are moved up and down within the alignment using the up and down arrow keys. Sequences are deleted from the alignment by selecting a sequence name and pressing delete/backspace on the keyboard. Specific regions of the sequence are altered by selecting the residue(s) in a red box by left clicking and dragging; the selected area is where the sequence alterations will occur. To delete the selected residues or gap characters, press delete/backspace. To realign a selected region, hold control on the keyboard while left clicking and dragging; you must select some gap characters when realigning sequence in order to have empty space to move the residues into. These operations will be used during the curation process. See Note: 20.

4. Identify and remove incomplete sequences by examining the alignment and looking for sequences that are not long enough to span the length of the alignment. Truncated sequences in the N and C termini should be removed. See Note: 21.

5. Inspect each column for gap characters in the overview window and delete and sequences that are missing crucial parts of the protein. The inclusion of a gap character at a position is the implicit acknowledgement of structural uncertainty at that position; in order for a position to be analyzed, sequences that contain gaps at that position must be deleted. See Note: 22.

6. Under the alignment, there is a heuristic for judging the quality of the alignment at that position, Local Covariation. Local Covariation identifies regions of a protein that are likely to be misaligned. Use Local Covariation as a guide to inspect positions in the protein (Figure

2.3).

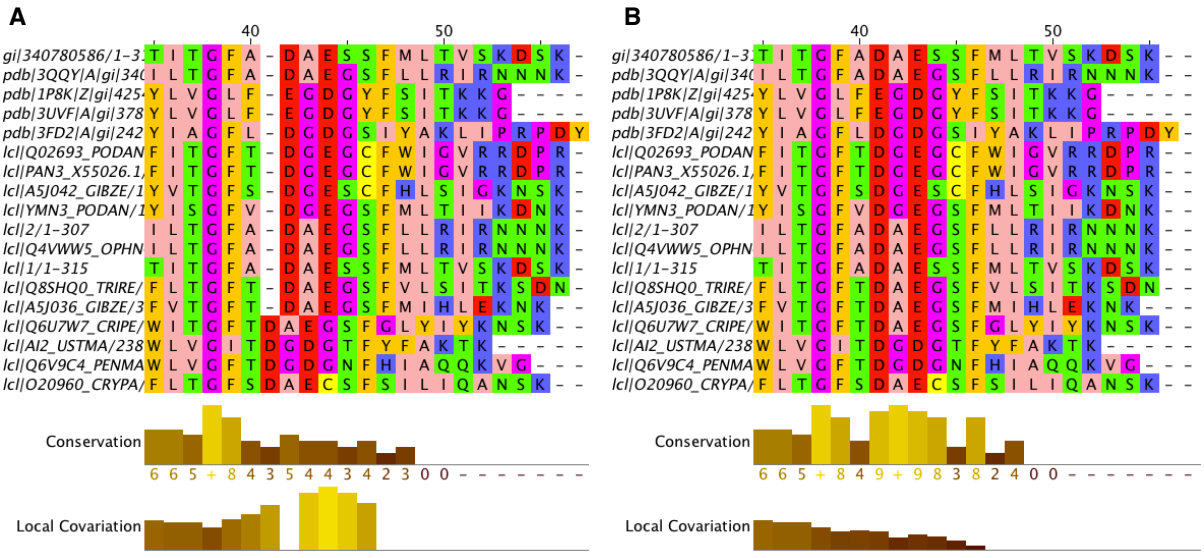

Figure 2.3: Realigning sequences using LoCo. Panel A: A screenshot of the LoCo workspace area which includes a misalignment. The misalignment can be identified by the high local covariation seen in the Local Covariation histogram below the sequence alignment itself. Panel B: A screenshot of the corrected alignment shows improved conservation, reduced local covariation, and improved alignment according to the Zappo colour scheme.

7. In each position that has high local covariation (a yellow-coloured bar) complete this procedure. Select the region of local covariation (which includes the contiguous positions with a yellow bar and 5 positions to the right) and select *Calculate* → *Calculate Tree* → *Neighbour Joining Using % Identity* from the menu bar. Then select *Calculate* → *Sort* → *by Tree Order*. Inspect your newly clustered region for positions that appear to be shifted out of alignment and realign them as outlined in step 3. If the realignment is correct, the local covariation score should go down. See Note: 23.

8. Another potential source of local covariation signal is the inclusion of paralogous sequences, sequences that are similar by because of shared ancestry, but have diverged due to a gene duplication event and now have a new function. Removing paralogous sequences can also decrease the local covariation score and increase downstream accuracy.

9. Finally, identify and remove any sequences that do not conform to known properties of the protein. For example, serine proteases should contain a serine in the catalytic position, or

LAGLIDADG homing endonucleases should contain acidic residues in the catalytic positions.

10. Restore the target sequence to the beginning of the alignment by selecting it and using the up arrow key to move it back to the top. Save a copy of this alignment.

11. Trim your alignment by selecting the column which contains the first residue in the target structure and click *Edit* → *Remove Left*. Trim the C-terminus by selecting the column containing the final residue in the target sequence and clicking *Edit* → *Remove Right*.

12. Remove redundant sequences by selecting the entire alignment (*Select* → *Select All*) and then clicking *Edit* → *Remove Redundancy*. Then adjust the redundancy threshold to *90* and then click *Remove*. Save this alignment; it will be used as input for the coevolution analysis. Be careful not to remove your target sequence when removing redundancy as it is required for ensuring that numbering is correct in section 2.2.3.5. It is crucial that at least 125 sequences are in the final alignment.

2.3.6 Coevolution analysis

1. Ensure that your alignment has enough sequences and ungapped positions to perform a coevolution analysis. At this point the alignment should contain at least 125 sequences and at least 50 ungapped positions.

2. Ensure the programs MIp.pl, MIp, and dist_pdb have been compiled and placed in the *$PATH* shell variable. You can verify this by typing each of the program names into your shell interface which will print out a help message for MIp.pl and a brief message from MIp and dist_pdb.

3. Run the following if you have a pdb file:

MIp.pl -i alignment.fasta -o coevolution.txt -d T -p pdb_file.pdb -e 0.001 -a 1

Run the following if you do not have a pdb file

MIp.pl -i alignment.fasta -o coevolution.txt -d F -e 0.001 -a 1. See Note: 24.

4. Verify that MIp.pl has executed successfully by examining the main output file, count file and three .dot files.

5. Open the MIpToolset output file and verify that the *aa_c* and *aa_d* columns correspond to the *PDB_i* and *PDB_j* columns respectively. These columns represent the residue identity at the assigned number in the sequence and structure. If these values do not align, this means that the sequence and structure begin numbering in two different places; not all proteins begin their numbering at *1*. Use the *-a* option to set the offset between the sequence and PDB numbering. See Note: 25.

2.3.7 Visualizing and interpreting results

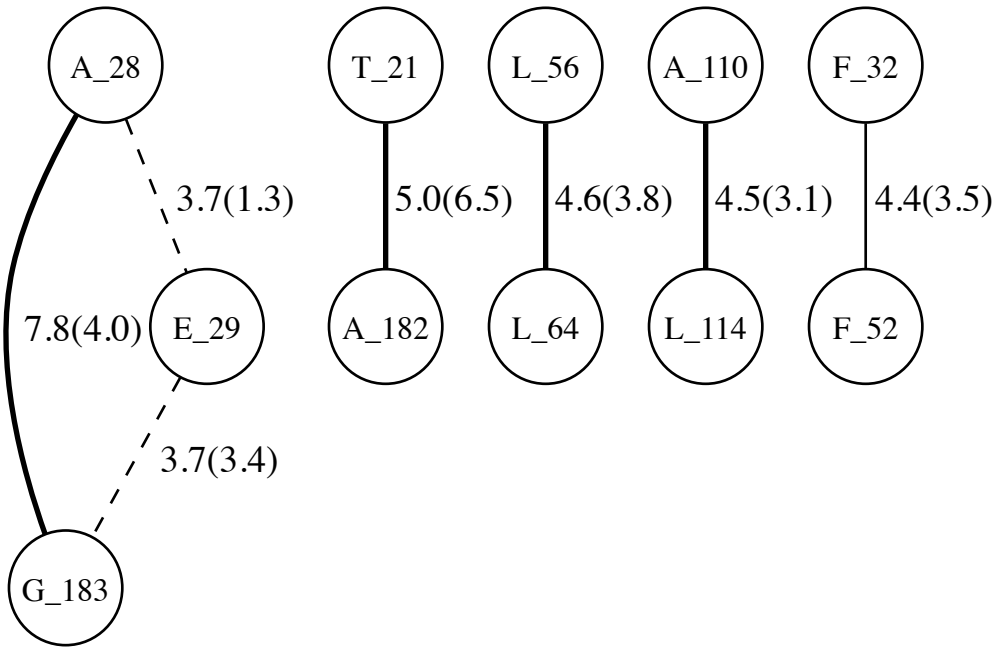

Figure 2.4: A network representation of coevolving residues. This network represents a small portion of a full network of coevolving residues from the LAGLIDADG Homing Endonuclease protein family. Each node in the network corresponds with a residue position in the sequence alignment, labeled by both residue number and identity. Each edge in the network represents a potential interaction via coevolution. Edge thickness corresponds to the coevolution score, (in this case the coevolution statistic *Zpx*). Edges are labeled by coevolution score, followed by inter-residue distance measured in angströms in parentheses.

1. The raw data is produced as a tab-delimited table so it is viewable in a standard spreadsheet program or statistical programming language. Excel users can append .xls to the end of their coevolution output (eg. coevolution.txt.xls) to make Excel interpret the file as an Excel Spreadsheet. The data can be explored by sorting in descending order in the *Zp*, *Zpx*, and *ndz* (which is ΔZp) columns. See Note: 26.

2. MIpToolset will create files with the extension .dot appended to the end of the given output file name. Open these files in *graphviz* to see the network of potentially coevolving residues as defined by the coevolution statistic. Labeled circles represent positions in the protein alignment numbered according to the target sequence/structure. Lines connecting circles represent covariation; line thickness indicates the strength of the coupling (Figure 2.4). See Note: 27.

3. Visualize the coverage and accuracy of your results by plotting the data in contact map format. Create an XY Scatter-style plot where the x and y axes are defined as positions in the protein and open circles represent positions less than 6 angströms apart in structure. Then plot smaller filled circles as the top-scoring pairs according to the selected coevolution statistic. The coevolution prediction coverage and correspondence with residue contacts provides an indication of the quality of the predictions by the coevolution statistic (Figure 2.5). See Note: 28

2.4 Notes

1. Windows users have a number of options for creating a Unix-like interface. One option is to install Linux on the Windows machine eg. Ubuntu (http://www.ubuntu.com/). Another option is to install Cygwin, a program that emulates a Unix-like interface without installing a new operating system (http://www.cygwin.com/). Both of these options work with this method, however a native unix-like environment is preferred and the Cygwin workaround is not supported. If you are going to use the Cygwin workaround, make sure that you install the sequence

analysis tools in a directory where Cygwin has the ability to read and write; also, make sure that Cygwin and the other bioinformatics tools are not blocked by your anti-virus software.

2. gcc is a program that builds software tools from source code. There are many ways to install gcc on your respective platform. For Mac OS X users, gcc is included as part of an add-on to the Xcode tool available on the Mac App Store, presently. Download Xcode from the App Store (https://itunes.apple.com/ca/app/xcode/id497799835); once Xcode is installed, the Command Line Tools are available from the Downloads section of Preferences. For Ubuntu users, type apt-get install gcc into a terminal window to download gcc. For those using the unsupported Windows Cygwin workaround, gcc is installed using the same setup.exe that installs Cygwin itself. Install gcc as is instructed for your respective operating system.

3. Your target sequence is ideally the wild-type sequence for your protein family of interest from the species of interest. In most cases this will be the human sequence, but in others it may be from the model organism you are studying or corresponds with a crystal structure you are interested in.

4. BLAST databases are split into multiple files, but addressed through the BLAST command line interface through the database name only. For example, the nr protein database is split into numbered files labeled nr.00.tar.gz, nr.01.tar.gz, nr.02.tar.gz. These files can be downloaded through your web browser, favourite FTP client, or through the update_blast.pl Perl script included in the aforementioned BLAST+ suite. Details are available from NCBI: ftp://ftp.ncbi.nlm.nih.gov/blast/documents/blastdb.html

5. Discriminating homology (sequences similar by common ancestry) and non-homologous analogy (sequences similar by convergence or coincidence, ie. homoplasy) is a major challenge of bioinformatics. It is critical to understand that sequence similarity and not sequence annotation must be used to collect homologues. Annotation errors or incomplete annotations will lead to an erroneous and incomplete final protein family alignment. The ubiquitous BLAST tool [1] can be used to infer homology by sequence similarity, but struggles to identify highly diverged homologues because it uses generic substitution matrices like BLOSUM [16]. PSI-

BLAST uses a similar search strategy, but builds position-specific scoring matrices tailored to the protein family over multiple iterations of searching [2]. The PSI-BLAST strategy improves detection of divergent homologues.

6. PSI-BLAST is similar to BLAST in that it is a software tool that searches for homologous protein sequences; however, PSI-BLAST uses an iterative search strategy which improves detection of sequentially dissimilar homologues. Briefly, PSI-BLAST employs a Position-Specific Scoring Matrix (PSSM) which defines how favourably an amino acid will be scored on a position-by-position basis as defined by the growing alignment of matches. PSI-BLAST performs multiple rounds of searching; an updated PSSM calculated and used each round. The fact that custom scoring matrices are used for each position in the protein rather than a generic scoring matrix generated by averaging across many positions in many proteins gives PSI-BLAST an advantage in detecting homology [2].

7. While the precise cutoff value for your PSI-BLAST may need to be revised, the starting point provided in 2.3.2.3 is a good heuristic. If the search returns too many dissimilar sequences (ie. paralogues), then make the search more strict. Conversely, if the search returns too few sequences, make the search less strict. To increase the speed of the search for the ideal cutoff, first adjust the E value by several orders of magnitude and observe the effect; much time is wasted gradually changing an E value by a factor of 2 or 3. Determining whether sequences are orthologous or paralogous is difficult and requires knowledge about the protein family. A lower score compared to the rest of the protein family may indicate homology with functional divergence. Other evidence includes 1) bidirectional top matches between two organisms, 2) including only one sequence per organism, and 3) the absence of critical functional residues may indicate alternate function and thus paralogous sequences.

8. If your protein family is well-studied, there may be an existing Conserved Domain [26] or PFAM [28] available as a starting point. Search the Conserved Domain Database (CDD) (http://www.ncbi.nlm.nih.gov/Structure/cdd/cdd.shtml) for your target protein family. If an entry exists, open the CDD structure alignment by expanding the *Structure* section, selecting

the maximum value for *Aligned Rows* and clicking the *Structure View* button. You will still need to perform the curation steps as even these database alignments can contain errors.

9. Cn3D uses a Position Specific Scoring Matrix (like PSI-BLAST) to indicate alignment quality and align sequences that do not have structural information. Each position in the alignment has a different scoring matrix which defines which residues are deemed favourable or unfavourable. The PSSM is calculated based on all the sequences included in the *Alignment Viewer*.

10. When choosing structures to import, the goal should be diversity. An attempt should be made to include structures which encompass the diversity of the protein family.

11. The strength of Cn3D as tool for creating protein family alignments is its ability to explicitly define regions of structural similarity and divergence. The underlying hypotheses of coevolution analysis are structural in nature, as inter-residue contact is used as a proxy for coevolution when benchmarking covariation statistics. Thus a structure-based view of homology is critical when creating alignments. The goal of creating a Cn3D alignment is to maximize the number of structurally-defined positions that are conserved among the entire protein family this is the function of the Block regions. Defining a position within a block, the aligner asserts that this position is structurally defined and required for membership in the protein family. Understandably, parameters of the protein family will dictate the number and length of each block region: a recently-derived, well-conserved, orthologous protein family will contain fewer, longer blocks, with fewer shorter gaps, while a more diverse and diverged family will contain fewer, shorter blocks and numerous longer gaps. As a general rule, the core of the protein should be easily defined in blocks as there is less structural divergence; conversely, segments of unstructured surface loops typically are not included in blocks, as they accept insertions and deletions, and are less likely to be structurally superimposable, and thus, defined. Structurally speaking, alignment within diverged gap regions is not meaningful because homology cannot be inferred. The Cn3D documentation advises that only regions within blocks are considered aligned; this is why positions that contain gaps are not included in a coevolution

analysis by the MIpToolset.

12. The structure alignment superimposition is defined entirely by the sequence alignment window. The structures are rigid and will be rotated and translated relative to one another in order to minimize the distance between alpha carbons of residues which occupy the same column in the alignment window. The menu command *File* → *Realign structures* must be selected every time you want the *Structure Viewer* window to be updated to reflect the current *Sequence Viewer* alignment. Normally, the structural superimposition reflects all positions in block regions in the sequence alignment; however, it is also possible to further restrict the structure alignment to highlighted/selected positions. This option is very useful in determining whether a segment of the alignment is structurally acceptable: an incorrectly aligned segment may appear to be aligned correctly because other correctly aligned positions will outvote the poorly aligned segment and create the illusion of correct structural superimposition between the rigid structures. To align based on a single segment, highlight it and answer Yes to the prompt; to align based on all block positions, answer No.

13. The placement of block regions is critical for generating a high quality alignment. While the blocks that are already defined in the Conserved Domain Database (CDD) [26] are generally very good, they are subject to human error and potential misalignments have been identified [11, 9] that could be attributed to human error in this important step: setting the blocks to reflect structural conservation. Blocks should extend across the conserved core of the protein, through any region that would not accept an insertion or deletion without drastically affecting the protein fold. Secondary structural elements should individually be included as a single block; although rarely alpha helices can accept a single residue insertion which will require a block to be split. Regions of structural divergence should not be included in blocks. These include surface loops, long unstructured linker regions, and sometimes the N- and C-termini. It is not uncommon to see a secondary structural element at a terminus need to be excluded from a block because a wild-type orthologue does not contain this structure.

14. It is possible for the majority of sequences to not merge because the structures available

in the structure alignment imply a block model that is too restrictive. If a conflict exists in a region of structural conservation where a gap is structurally possible, though not observed, consider splitting the block to allow the small insertion or deletion in that region. Likewise, watch out for structured, though non-critical, N- and C-terminal features which can be excluded from the block model; though, do not delete important features at either terminus because partial sequences will not merge. As the curator, it is your job to make these important interpretations for your protein family.

15. With each iteration, the PSSM encompasses more diversity within the protein family and (when done properly) includes fewer errors. Multiple iterations are required because dissimilar members of the protein family are likely misaligned in the initial iterations, but will be correctly placed once the PSSM represents the entire protein family accurately.

16. While it is still possible to obtain a high quality set of coevolution predictions using an alignment built from sequence alone, it is more difficult than using a structure based alignment. A greater emphasis will need to be spent on step 2.2.3.4, Curation, if step 2.2.3.3b is used instead of 2.2.3.3a. As well, it is difficult to evaluate the accuracy of the predictions; the standard benchmark is to use the fraction of pairs of covarying positions above a threshold that are in contact as a proxy for true coevolution. Some even infer indirect coevolution between non-contacting pairs if other comparably scoring pairs are in contact [4]. Without structural information, this evaluation is impossible.

17. For some users, an example file will automatically open many more windows than is necessary for LoCo to function on an unrelated example protein family. This will happen every time upon start up unless it is disabled. Uncheck the box located in *Tools* → *Preferences* → *Visual* → *Open File* to disable this example file upon start up of the tool. Close and reopen the program.

18. It is possible to get a holistic view of the alignment by using the *Overview Window*. This window provides rapid navigation by clicking on the region you wish to travel to. As well, it provides a way of looking for alignment abnormalities without scrolling through the

entire alignment in detail. Some problems will be more apparent more quickly in the *Overview Window*. Look for sequences with long gaps or abnormal gap placement. As well, look for sequences that do not fit with the coloured motifs of the other sequences. A sequence that looks like it doesnt belong likely contains a shift error or is not homologous. The ideal setup for curating an alignment is across two monitors: 1 monitor for editing the alignment and one for examining the entire alignment in the *Overview Window*.

19. If the *Local Covariation* bar is empty for the entire length of your alignment, this may be an indication that the MIp program did not build properly or does not have write permissions in its current directory. This is especially problematic for (unsupported) Windows users who are emulating a Unix-like interface using Cygwin. Consult the known_issues.txt and README files at (http://sourceforge.net/projects/locoprotein/files/) for more assistance. As well, make sure that your target (ie. first) sequence does not contain any non-canonical amino acids, as this can affect numbering. The easy solution to this problem is to use the up and down arrow keys to shift a different sequence into the top (target) position temporarily. (Remember that you can use the arrow keys to shift sequences up and down but never left or right; this will cause the entire sequence to shift visually and cause alignment problems).

20. Do not grow too attached to the sequences in your alignment. Do not hesitate to delete the ones that are incomplete or seem otherwise erroneous. Incorrect sequences are sources of error in downstream analysis.

21. Often, pre-computed alignments in major databases contain too many gaps to be used for coevolution analysis. For example the Full PFAM alignments for the single-chain two-domain LAGLIDADG homing endonuclease (PF00961) contains sequences that only cross one of the two domains. Columns that contain gaps are not included in the coevolution analysis.

22. Remember that coevolution scores are not calculated on gapped positions. When re-aligning a segment of an alignment to reduce the *Local Coevolution* score, be careful you do not simply erase the signal by introducing new gaps. Look for alternate alignments that are supported by sequence, structure, and *Local Covariation*-based evidence.

23. Some PDB files contain additional formatting options which are not parsed properly by the MIpToolset. If PDB and alignment residue identities do not match, search the PDB file for missing lines or additional *HETATM* lines interspersed throughout the *ATOM* definition lines.

24. Positions that contain gaps are excluded from analysis by MIp; be sure to inspect the FASTA alignment file. There are many reasons why the distance information does not match the alignment file (as indicated in by a mismatch between the residues in the coevolution output file). PDB files may omit or change some residues which are present in the wild-type sequence. They also may include additional *HETATM* lines mid-chain, or label *ATOMS* as *HETATMs*. You may get different results depending on where you obtain the PDB file as well; experience has shown that PDB files obtained from the RCSB Protein Data Bank may be slightly different than PDB files obtained from NCBI. As well, be sure that the first sequence in the FASTA alignment is the same as the structure input as a PDB.

25. MIp is approximately normal if generated on random sequence, which is why Z scores are used for MIp. Do not misunderstand Z scores as significance the wrong way. Zpx scores are comparable, though slightly more accurate. ΔZp is a heuristic and does not have a rigorous statistical framework.

26. A network representation of covarying positions is an excellent way of gaining a holistic view of the potentially coevolving positions. Every node in the network represents a position in the protein numbered according to the target sequence. Every edge (line) in the graph corresponds with a comparatively high covariation score; thicker lines represent stronger coupling. The actual covariation score and distance label each edge ie. 4.5(5.0) represents a pair with a covariation score of 4.5 and a distance of 5.0 angstroms between the two closest non-hydrogen atoms of the respective residues in the target structure. If no structure data is specified, the distances will appear as 0.0. Interestingly, some covarying pairs may be in contact between protein chains; intra-chain distances will indicate that the positions are not in contact, but will appear in contact when viewing the quaternary structure of the protein.

27. Contact Maps are an excellent way to view the relationship between covarying posi-

tions and the proteins secondary and tertiary structure. Here we define contact as any non-hydrogen atom less than six angstroms apart. In a contact map, the X and Y axes represent the position in the target sequence; thus the points that run along the diagonal of the plot represents sequence-local interactions and off-diagonal points represent sequence-distant interactions. Contact always occurs along the diagonal because a protein is a single chain. But other common interactions are visible in the contact map as well, like interactions between termini, which manifest as contacts in the furthest corners of the contact map. Furthermore, parallel interactions, like parallel beta-sheet, will appear as off-diagonal contacts that run parallel to the diagonal; anti-parallel interactions, like anti-parallel helices, will appear as contacts that run anti-parallel to the diagonal.

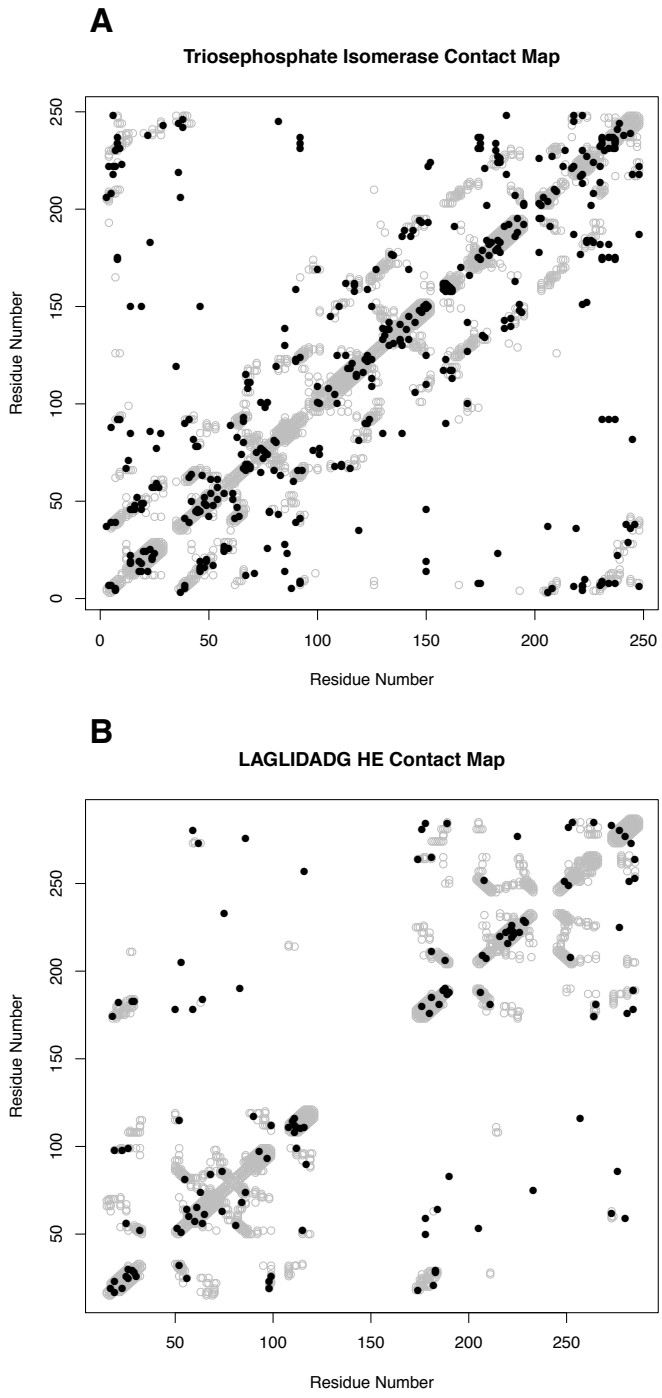

Figure 2.5: Contact Map representations of coevolution predictions. Panel A. Predicted contact map visualization of a coevolution network. Contacting residues are labeled grey. Predicted potentially coevolving residues (*Zpx* greater than 3) are printed on top of contacting residues and coloured black. The correspondence between contact and predicted coevolving pair is apparent in this visualization. Panel A represents near-optimal results obtained from analyzing a Triosephosphate Isomerase alignment with optimal characteristics. Panel B represents typical-use results obtained from a LAGLIDADG HE alignment with fewer sequences, and a less-confident alignment.

Bibliography

[1] S F Altschul, W Gish, W Miller, E W Myers, and D J Lipman. Basic local alignment search tool. *Journal of Molecular Biology*, 215(3):403–410, October 1990.

[2] S F Altschul, T L Madden, A A Schaffer, J Zhang, Z Zhang, W Miller, and D J Lipman. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Research*, 25(17):3389–3402, September 1997.

[3] Robert B Ash. *Information Theory*. Courier Dover Publications, 1965.

[4] Lukas Burger and Erik van Nimwegen. Disentangling direct from indirect co-evolution of residues in protein alignments. *PLoS Computational Biology*, 6(1):e1000633, January 2010.

[5] Michele Clamp, James Cuff, Stephen M Searle, and Geoffrey J Barton. The Jalview Java alignment editor. *Bioinformatics*, 20(3):426–427, February 2004.

[6] T M Cover and Joy A Thomas. *Elements of information theory*. New York, 1991.

[7] R J Dickson and G B Gloor. The MIp Toolset: an efficient algorithm for calculating Mutual Information in protein alignments. *arXiv.org*, 2013.

[8] RJ Dickson, LM Wahl, AD Fernandes, and GB Gloor. Identifying and seeing beyond multiple sequence alignment errors using intra-molecular protein covariation. *PLoS ONE*, 5(6):e11082, 2010.

[9] Russell J Dickson and Gregory B Gloor. Protein sequence alignment analysis by local covariation: coevolution statistics detect benchmark alignment errors. *PLoS ONE*, 7(6):e37645, 2012.

[10] Russell J Dickson and Gregory B Gloor. The MIp Toolset: an efficient algorithm for calculating Mutual Information in protein alignments. *arXiv preprint arXiv:1304.4573*, 2013.

[11] Russell J Dickson, Lindi M Wahl, Andrew D Fernandes, and Gregory B Gloor. Identifying and seeing beyond multiple sequence alignment errors using intra-molecular protein covariation. *PLoS ONE*, 5(6):e11082, 2010.

[12] S D Dunn, L M Wahl, and G B Gloor. Mutual information without the influence of phylogeny or entropy dramatically improves residue contact prediction. *Bioinformatics*, 24(3):333–340, January 2008.

[13] Sean R Eddy. Accelerated Profile HMM Searches. *PLoS Computational Biology*, 7(10):e1002195, October 2011.

[14] Robert C Edgar. MUSCLE: multiple sequence alignment with high accuracy and high throughput. 32(5):1792–1797, 2004.

[15] John Ellson, Emden Gansner, Lefteris Koutsofios, Stephen C North, and Gordon Woodhull. Graphviz—open source graph drawing tools. *Lecture Notes in Computer Science*, pages 483–484, 2002.

[16] 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(22):10915–10919, November 1992.

[17] C W Hogue. Cn3D: a new generation of three-dimensional molecular structure viewer. *Trends Biochem Sci*, 22(8):314–316, August 1997.

[18] Alexander Kawrykow, Gary Roumanis, Alfred Kam, Daniel Kwak, Clarence Leung, Chu Wu, Eleyine Zarour, Phylo players, Luis Sarmenta, Mathieu Blanchette, and Jérôme Waldispühl. Phylo: a citizen science approach for improving multiple sequence alignment. *PLoS ONE*, 7(3):e31362, 2012.

[19] Kazuharu Misawa Kei-ichi Kuma Takashi Miyata Kazutaka Katoh. MAFFT: a novel method for rapid multiple sequence alignment based on fast Fourier transform. *Nucleic Acids Research*, 30(14):3059, July 2002.

[20] Firas Khatib, Frank Dimaio, Seth Cooper, Maciej Kazmierczyk, Miroslaw Gilski, Szymon Krzywda, Helena Zabranska, Iva Pichova, James Thompson, Zoran Popović, Mariusz Jaskolski, and David Baker. Crystal structure of a monomeric retroviral protease solved by protein folding game players. *Nature Structural & Molecular Biology*, pages 1–3, September 2011.

[21] M Kimura. Evolutionary rate at the molecular level. *Nature*, 217(5129):624–626, 1968.

[22] M Kimura and T Ota. On some principles governing molecular evolution. *Proc Natl Acad Sci U S A*, 71(7):2848–2852, 1974.

[23] Benjamin P Kleinstiver, Andrew D Fernandes, Gregory B Gloor, and David R Edgell. A unified genetic, computational and experimental framework identifies functionally relevant residues of the homing endonuclease I-BmoI. *Nucleic Acids Res*, 2010.

[24] Yuheng Li, Nicholas Chia, Mario Lauria, and Ralf Bundschuh. A performance enhanced PSI-BLAST based on hybrid alignment. *Bioinformatics*, 27(1):31–37, January 2011.

[25] Ari Löytynoja and Nick Goldman. Phylogeny-aware gap placement prevents errors in sequence alignment and evolutionary analysis. *Science*, 320(5883):1632–1635, June 2008.

[26] Aron Marchler-Bauer, John B Anderson, Farideh Chitsaz, Myra K Derbyshire, Carol DeWeese-Scott, Jessica H Fong, Lewis Y Geer, Renata C Geer, Noreen R Gonzales,

Marc Gwadz, Siqian He, David I Hurwitz, John D Jackson, Zhaoxi Ke, Christopher J Lanczycki, Cynthia A Liebert, Chunlei Liu, Fu Lu, Shennan Lu, Gabriele H Marchler, Mikhail Mullokandov, James S Song, Asba Tasneem, Narmada Thanki, Roxanne A Yamashita, Dachuan Zhang, Naigong Zhang, and Stephen H Bryant. CDD: specific functional annotation with the Conserved Domain Database. *Nucleic Acids Res*, 37(Database issue):D205–10, 2009.

[27] Eyal Privman, Osnat Penn, and Tal Pupko. Improving the performance of positive selection inference by filtering unreliable alignment regions. *Molecular Biology and Evolution*, 29(1):1–5, January 2012.

[28] Marco Punta, Penny C Coggill, Ruth Y Eberhardt, Jaina Mistry, John Tate, Chris Boursnell, Ningze Pang, Kristoffer Forslund, Goran Ceric, Jody Clements, Andreas Heger, Liisa Holm, Erik L L Sonnhammer, Sean R Eddy, Alex Bateman, and Robert D Finn. The Pfam protein families database. *Nucleic Acids Res*, 40(Database issue):D290–301, January 2012.

[29] R Core Team. R: A Language and Environment for Statistical Computing. 2013.

[30] Johannes Söding. Protein homology detection by HMM-HMM comparison. *Bioinformatics*, 21(7):951–960, April 2005.

[31] E L Sonnhammer, S R Eddy, and R Durbin. Pfam: a comprehensive database of protein domain families based on seed alignments. *Proteins: Structure, Function, and Bioinformatics*, 28(3):405–420, July 1997.

[32] G Talavera and J Castresana. Improvement of phylogenies after removing divergent and ambiguously aligned blocks from protein sequence alignments. *Systematic biology*, 56(4):564, 2007.

[33] Andrew M Waterhouse, James B Procter, David M A Martin, Michèle Clamp, and Geoffrey J Barton. Jalview Version 2–a multiple sequence alignment editor and analysis workbench. *Bioinformatics*, 25(9):1189–1191, May 2009.

Chapter 3

The MIp Toolset Algorithm

3.1 Abstract

Background: Coevolution within a protein family is often predicted using statistics that measure the degree of covariation between positions in the protein sequence. Mutual Information is a measure of dependence between two random variables that has been used extensively to predict intra-protein coevolution.

Results: Here we provide an algorithm for the efficient calculation of Mutual Information within a protein family. The algorithm uses linked lists which are directly accessed by a pointer array. The linked list allows efficient storage of sparse count data caused by protein conservation. The direct access array of pointers prevents the linked list from being traversed each time it is modified.

Conclusions: This algorithm is implemented in the software MIpToolset, but could also be easily implemented in other Mutual Information based standalone software or web servers. The current implementation in the MIpToolset has been critical in large-scale protein family

A version of this chapter has been published.
RJ Dickson, GB Gloor. (2013). The MIp Toolset: an efficient algorithm for calculating Mutual Information in protein alignments. arXiv:1304.4573

analysis and real-time coevolution calculations during alignment editing and curation.

The MIpToolset is available at:

https://sourceforge.net/projects/miptoolset/

3.2 Introduction

The identification and analysis of covarying positions in a protein family gives important insights into that family's evolutionary history and provides information about sites that are important for function and structural stability as it is believed that covariation implies coevolution [1, 13, 8, 14]. Coevolutionary analysis of protein families is important because it potentially provides a direct link between primary sequence, in the form of multiple sequence alignments, and structure/function predictions. Covariation between positions in a protein family is assumed to derive from phylogenetic, structural, functional, interaction, and stochastic signals [1]. Decomposing this signal is difficult because the phylogenetic and stochastic signal can overwhelm the structural and functional signal [12]. Furthermore, alignment errors have been shown to produce misleading erroneous signal [4].

One of the most popular methods for quantifying covariation in proteins is Mutual Information (MI). There are many coevolution prediction methods which are derived from MI [8, 6, 11, 2, 4]. As well, there are many web-based servers which will calculate Mutual Information from a submitted protein alignment [10, 3, 16, 9]. Despite its simple formulation, calculation of MI is computationally demanding, largely because it must be calculated for all pairs of positions in the alignment, meaning it scales n^2 relative to the length of the alignment. Further, calculating inter-protein coevolution requires concatenated alignments which increases the effective number of pairs of positions.

Herein we describe an algorithm for calculating MI in protein alignments with high efficiency. This algorithm allows for database-wide analysis [4] and real-time calculation of covariation during alignment curation [5]. This algorithm is included as part of the MIpToolset.

3.3 Algorithm

3.3.1 Mutual Information

The calculation and formulation of Mutual Information is described in detail in [12]; it is outlined here to provide necessary background to understand the optimizations of the MIpToolset algorithm.

Mutual Information measures the degree of covariation between two random variables (in our case, protein alignment positions X and Y) using the Information Theoretic quantity Entropy (H).

$$MI_{x,y} = H_x + H_y - H_{x,y} \tag{3.1}$$

Information Entropy (H) can be understood as the measure of uncertainty of the identity of the amino acid at some position x. As shown in equation 3.2, the Entropy (H) for position x is calculated using the probability of each of the 20 amino acids appearing at that position. Since the actual probabilities are unknown, the amino acid frequencies in the input alignment are used to approximate these values.

$$H_x = -\sum_{i=1}^{20} p(x_i) \log_{20} p(x_i) \tag{3.2}$$

The MI between positions X and Y is the sum of the Entropy of each position minus the "joint Entropy" between them. The enumeration of joint entropy is the rate-limiting step of Mutual Information calculations. Joint Entropy is calculated similarly to Entropy, but it involves the calculation of probability of all pairs of amino acids that occur between position x and position y (Equation 3.3).

$$H_{x,y} = -\sum_{i=1}^{20} \sum_{j=1}^{20} p(x_i, y_j) \log_{20} p(x_i, y_j) \tag{3.3}$$

The naïve calculation of joint entropy is inefficient because it involves populating a 20 x

20 matrix for every pair of amino acids found for every pair of positions. This is a 400-entry matrix for n^2 positions. This approach, while easy to implement, uses an unnecessary amount of memory as it does not exploit the fact that most positions will be moderately conserved and, thus, most positions will have a value of zero in the joint entropy count matrix.

3.3.2 Storage of sparse matrix in linked list

It is worth noting that calculation of *MI* involves two types of "pairs": Pairs of *positions*, which represent the homologous 'columns' in a protein family multiple sequence alignment (MSA), and pairs of *amino acids*, which are the corresponding entries from a pair of positions within a single sequence. So a pair of positions, might be position 10 and position 45 within a protein sequence; at this pair of positions, there will be many amino acid pairs corresponding to the identity of the amino acids at positions 10 and 45 in the sequence (ie. DF, LS, DH etc.).

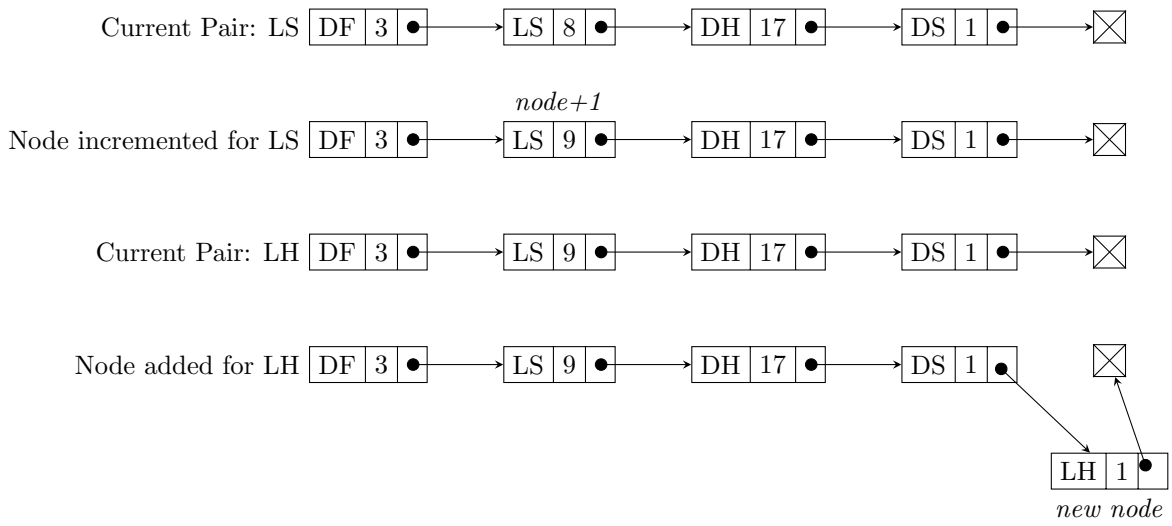

Figure 3.1: Linked list storage of amino acid pair counts. This figure demonstrates how two new amino acid pairs, LS and LH, are added to the growing linked list data structure which stores amino acid pair counts. First, LS is added to the list by incrementing the existing LS node. Second, the pair LH is added to the list by creating a new node labeled LH and adding it to the list with counter set to 1.

A straightforward way to store the counts between positions x and y is to use a linked list

data structure (Figure 3.1). Each node in the linked list stores two values for the calculation of Joint Entropy, the identity of the amino acid pair, and the respective count. Each node also contains a pointer to the next node in the list, or *null* if the node is the terminal node.

The program iterates over the protein alignment, enumerating the amino acid pairs, just as it would if it were in the naïve implementation. If an entry in the linked list exists for a given amino acid pair, the node's counter is incremented. If no such entry exists a new node is appended to the end of the list for that amino acid pair. This list can be traversed efficiently as these counts will be used for future calculations. This efficient storage makes it possible to efficiently analyze very long alignments.

3.3.3 Direct access to linked list improves speed

The limitation of the linked list storage method, if a linked list is used on its own, is that the list will need to be traversed each time a node is to be updated or created to check whether that pair exists in the data structure. This challenge can be overcome by using an array of pointers to linked list nodes. The disadvantage of the linked list storage solution is that it lacks "direct access" provided by a two-dimensional array from the naïve implementation. By combining the two, it is possible to achieve a "best of both worlds" solution.

A single "direct-access" 20 x 20 array is created, with the nodes in the array corresponding to the 400 possible amino acid pairs (Figure 3.2). When an amino acid pair is encountered by the main count enumeration loop, the direct-access array is checked. If the entry for that pair is *null*, then a new linked node is appended to the end of the growing linked list for that pair of positions with a count of 1; next, the entry in the direct-access matrix is set as a pointer to the newly created linked list node.

Conversely, if the entry corresponding to the amino acid pair contains a pointer, the program follows the pointer to the corresponding linked list node and increments the counter by 1. After the two positions have been fully enumerated, all entries in the direct-access array are reset to *null* and it can be reused. Thus, the direct-access array strategy maintains the advantages of a

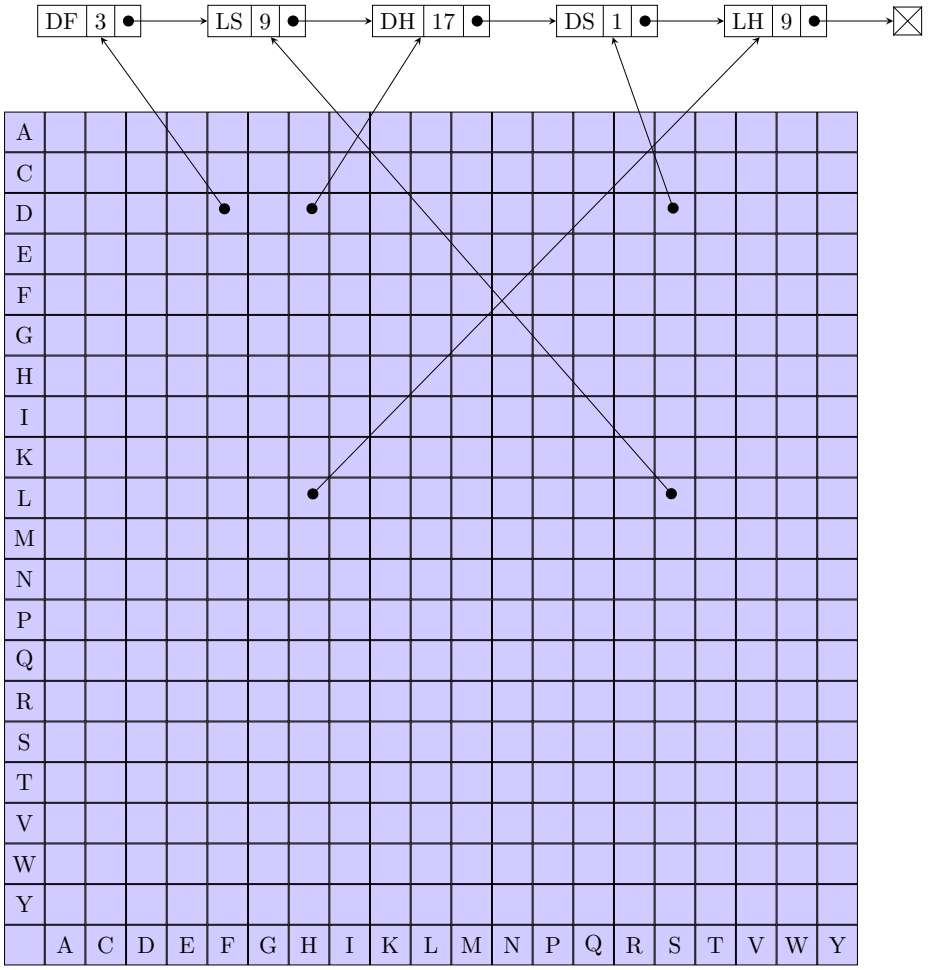

Figure 3.2: Direct-access array of pointers to growing linked list. This figure demonstrates how the direct-access array provides instant access to any part of the linked list without the need to traverse the list. This array is only allocated once and can be reused for each pair of positions.

linked list storage solution without the disadvantage of needing to traverse the list every time, at the trade-off cost of only 400 pointers.

3.3.4 Integration in the MIpToolset

This algorithm has been included as part of the MIpToolset, a collection of C- and Perl-based programs which calculate covariation statistics and inter-residue distances from protein alignments and databases. A full description of sequence collection and alignment is available in (Dickson and Gloor, Methods Mol. Biol. 2013 *submitted*).

In brief, the input to the program is a protein alignment containing more than 150 sequences less than 90% identical and containing more than 50 ungapped positions. It is recommended that the alignment be manually analyzed by the investigator to ensure the alignment does not contain errors which will lead to false-positive results [4, 5]. For example, the curation tool LoCo [5], based on the alignment viewer Jalview [15], provides a visualization of the likely-misaligned regions of the alignment. The program also optionally accepts a PDB structure corresponding to a sequence in the protein family. This structure is used to generate inter-residue distances which are commonly used to validate coevolution predictions.

The output of the program is a large list of pairs of positions and their corresponding co-variation statistics. The MIpToolset presently generates Mutual Information, as well as several more accurate derivations including MIp (and its normalized counterpart Zp) [6], Zpx and ΔZp [4]. A coevolution network file is also produced which can be visualized using Graphviz [7].

3.4 Conclusions

It is established that MI by itself is not particularly accurate in predicting coevolving positions because it correlates with Entropy [12], misleading phylogenetic signal [6], and alignment errors [4]. Furthermore, analyzing gaps as the "21st amino acid" causes misleading results which is partially why the aforementioned studies excluded positions containing gaps from the analysis (Dickson et al. submitted). It is possible to overcome some limitations of raw MI by using various corrections to MI [6, 11, 2, 4]. Typically these corrections based on an analysis of raw MI values are computationally inexpensive and so heavy optimization is not necessary. Thus the algorithm and software described herein can be used to reduce the time and memory required to calculate most MI-derived statistics.

The MIpToolset has been tested on Unix-like operating systems and is implemented in C for efficiency, with a Perl wrapper for handling input/output issues. The speed and efficiency

of the MIpToolset has allowed for efficient database-wide analysis [4] and the detection of protein family misalignments using an MI-derived method in real-time as the user edits their alignment in the software tool LoCo [5]. To our knowledge, this is the fastest implementation of the coevolution statistics MIp, Zp, Zpx, and ΔZp[6, 4].

It is available at: https://sourceforge.net/projects/miptoolset/

Bibliography

[1] W R Atchley, K R Wollenberg, W M Fitch, W Terhalle, and A W Dress. Correlations among amino acid sites in bHLH protein domains: an information theoretic analysis. *Mol Biol Evol*, 17(1):164–178, January 2000.

[2] Cristina Marino Buslje, Javier Santos, Jose Maria Delfino, and Morten Nielsen. Correction for phylogeny, small number of observations and data redundancy improves the identification of coevolving amino acid pairs using mutual information. *Bioinformatics*, 25(9):1125–1131, May 2009.

[3] Abhijit Chakraborty, Sapan Mandloi, Christopher J Lanczycki, Anna R Panchenko, and Saikat Chakrabarti. SPEER-SERVER: a web server for prediction of protein specificity determining sites. *Nucleic Acids Res*, 40(Web Server issue):W242–8, July 2012.

[4] RJ Dickson, LM Wahl, AD Fernandes, and GB Gloor. Identifying and seeing beyond multiple sequence alignment errors using intra-molecular protein covariation. *PLoS ONE*, 5(6):e11082, 2010.

[5] Russell J Dickson and Gregory B Gloor. Protein sequence alignment analysis by local covariation: coevolution statistics detect benchmark alignment errors. *PLoS ONE*, 7(6):e37645, 2012.

[6] S D Dunn, L M Wahl, and G B Gloor. Mutual information without the influence of phylogeny or entropy dramatically improves residue contact prediction. *Bioinformatics*, 24(3):333–340, January 2008.

[7] John Ellson, Emden Gansner, Lefteris Koutsofios, Stephen C North, and Gordon Woodhull. Graphviz—open source graph drawing tools. *Lecture Notes in Computer Science*, pages 483–484, 2002.

[8] Gregory B Gloor, Louise C Martin, Lindi M Wahl, and Stanley D Dunn. Mutual information in protein multiple sequence alignments reveals two classes of coevolving positions. *Biochemistry*, 44(19):7156–7165, May 2005.

[9] Rodrigo Gouveia-Oliveira, Francisco S Roque, Rasmus Wernersson, Thomas Sicheritz-Ponten, Peter W Sackett, Anne Mølgaard, and Anders G Pedersen. InterMap3D: predicting and visualizing co-evolving protein residues. *Bioinformatics*, 25(15):1963–1965, August 2009.

[10] Dániel Kozma, István Simon, and Gábor E Tusnády. CMWeb: an interactive on-line tool for analysing residue-residue contacts and contact prediction methods. *Nucleic Acids Res*, 40(Web Server issue):W329–33, July 2012.

[11] Daniel Y Little and Lu Chen. Identification of Coevolving Residues and Coevolution Potentials Emphasizing Structure, Bond Formation and Catalytic Coordination in Protein Evolution. *PLoS ONE*, 4(3):e4762, March 2009.

[12] L C Martin, G B Gloor, S D Dunn, and L M Wahl. Using information theory to search for co-evolving residues in proteins. *Bioinformatics*, 21(22):4116–4124, 2005.

[13] ERM Tillier and TWH Lui. Using multiple interdependency to separate functional from phylogenetic correlations in protein alignments. *Bioinformatics*, 19(6):750–755, 2003.

[14] Simon A A Travers and Mario A Fares. Functional coevolutionary networks of the Hsp70-Hop-Hsp90 system revealed through computational analyses. *Mol Biol Evol*, 24(4):1032–1044, 2007.

[15] Andrew M Waterhouse, James B Procter, David M A Martin, Michèle Clamp, and Geoffrey J Barton. Jalview Version 2–a multiple sequence alignment editor and analysis workbench. *Bioinformatics*, 25(9):1189–1191, May 2009.

[16] Kevin Y Yip, Prianka Patel, Philip M Kim, Donald M Engelman, Drew McDermott, and Mark Gerstein. An integrated system for studying residue coevolution in proteins. *Bioinformatics*, 24(2):290–292, January 2008.

Chapter 4

The alignment-coevolution relationship

4.1 Abstract

4.1.1 Motivation

The statistical non-independence of protein alignment positions reveals useful structure and functional information about a protein family. However, statistics that detect such coevolution within proteins are easily misinterpreted. The purpose of this work is to provide a mental framework for intuiting the fundamentals of Mutual Information-based covariation statistics and why caution should be used when attempting to infer coevolution between gapped positions.

4.1.2 Results

We clarify the relationship between intra-protein coevolution and the covariation statistics used to infer it; we also demonstrate the relationship between alignment and covariation as it relates

A version of this chapter has been submitted for publication

RJ Dickson, GB Gloor. (2013). Gambling on Gaps: An Explanation of the Alignment-Coevolution Relationship. Bioinformatics. Under review.

to gaps, while highlighting caveats in analysis using illustrations and experiment. We provide demonstrations of how gapped positions can mislead covariation statistics, yielding false-positive results. In a database-wide analysis, we demonstrate that contact prediction within gap regions is not effective. We conclude that when conducting coevolution analyses, gaps should not be treated as the 21st character.

4.2 Introduction

Protein sequence analysis methods are not "plug-and-play", despite the fact that they are often treated as such. Tools that statistically analyze sequence alignments contain many assumptions about the nature of the alignment and how it was created. When used incorrectly, these tools return misleading results, often with no warning to the user. Our goal is to provide guidance for using bioinformatic tools to prevent misinterpretation of sequence data when using tools developed to detect molecular covariation in protein sequence alignments. More specifically, we will address the debate about insertions and deletions in sequence alignments when calculating protein covariation. We demonstrate why gaps should not be interpreted as the 21st residue and furthermore why caution should be used when analyzing any alignment position that contains a gap character or that is on the margins of the gap.

Coevolution within a protein family is detected using many different methods, all of which contain explicit and implicit assumptions about the nature of the protein alignment being analyzed. Some of these assumptions are explicitly stated in the tools' respective publications and manuals, eg. how numerous and diverse do the sequences in the alignment have to be? Other assumptions underlying coevolution methods are less obvious as they are rooted in an implicit biochemical framework, eg. what is a gap and how should it be interpreted? Though it is not necessarily obvious, violating an implicit biochemical assumption can result in the same erroneous results as violating a simple explicit requirement like not including enough sequences in the input alignment.

The generation of a molecular coevolution dataset begins with making a high quality multiple sequence alignment. Parameter choices and the assumptions behind them that are included in this step are easy to overlook. Thus, it is often not appreciated that gapped positions in protein families are usually placed arbitrarily by the alignment algorithms themselves. The standard approach used by programs such as MUSCLE [9], CLUSTALW [27], or T-Coffee [20] is to partition residues in gaps equally between the two gap ends. Some tools, such as Cn3D [12], explicitly acknowledge this and allow the user to place residues in gaps according to whim. Still others, such as PRANK [15, 16] place gaps at evolutionarily informative positions. Whichever method is chosen, this simple partitioning violates the standard assumptions of positional homology implicit in sequence alignment and will have an obvious effect on downstream analyses.

It is thus critically important to develop an intuition about the relationship between the biological mechanism of coevolution, the assumptions and methods used to generate the alignments, and the statistical methods used to find it called covariation. In the same way that positional conservation is thought to indicate residue importance, positional covariation is thought to indicate an important structural or functional interaction [1].

4.3 Methods

4.3.1 Synthetic alignment demonstration

Figure 4.1 is a synthetic alignment generated for instructive purposes using LoCo [7], a modified version of Jalview [30]. The synthetic alignment was designed to highlight the archetypal positions and positional relationships including strong coevolution, perfect conservation, and random assortment. The full alignment from which the coevolution calculations are made is panel is 500 sequences long; a small subset of the sequences is shown for the demonstration.

The *Zp* coevolution scores were calculated using the MIpToolset [5]. The heatmap plot below the alignment was created using R [22].

4.3.2 Correspondence of gap positions and covarying pairs shown across alignment methods

Figure 4.2 was created by realigning the seed alignment of PF00112 from PFAM [21], using CLUSTALW [27], MUSCLE [9], and PRANK [15, 16] and removing positions that were not included in the target sequence, CPR6_CAEEL, selected arbitrarily; Zp (normalized MIp) scores were calculated according to [8, 6] and plotted in R [22].

4.3.3 Database-wide covariation accuracy analysis

Figure 4.3 was generated from a dataset of 314 alignments (Supplementary Table 1) selected out of all multiple sequence alignments in the CDD database, but removing alignments which did not have an associated structure, had fewer than 125 sequences, were shorter than 50 positions, had file format or numbering abnormalities or conflicts with the structural data. The MIp-Toolset [5] was modified to allow gap characters to be included in the analysis; as demonstrated in this paper, this is not recommended for general-purpose use. MIp scores for each alignment were separated by their location relative to gaps in the alignment. Positions within the same contiguous gap formed one class and positions that contained no gap characters formed another. These pairs of positions were ranked for each alignment and the mean fraction of pairs in contact for each class were plotted. This type of plot is an established way of benchmarking the accuracy of coevolution methods; in this case, it is used to compare the quality of the alignment positions, pertaining to their location in the alignment, rather than the coevolution methods themselves.

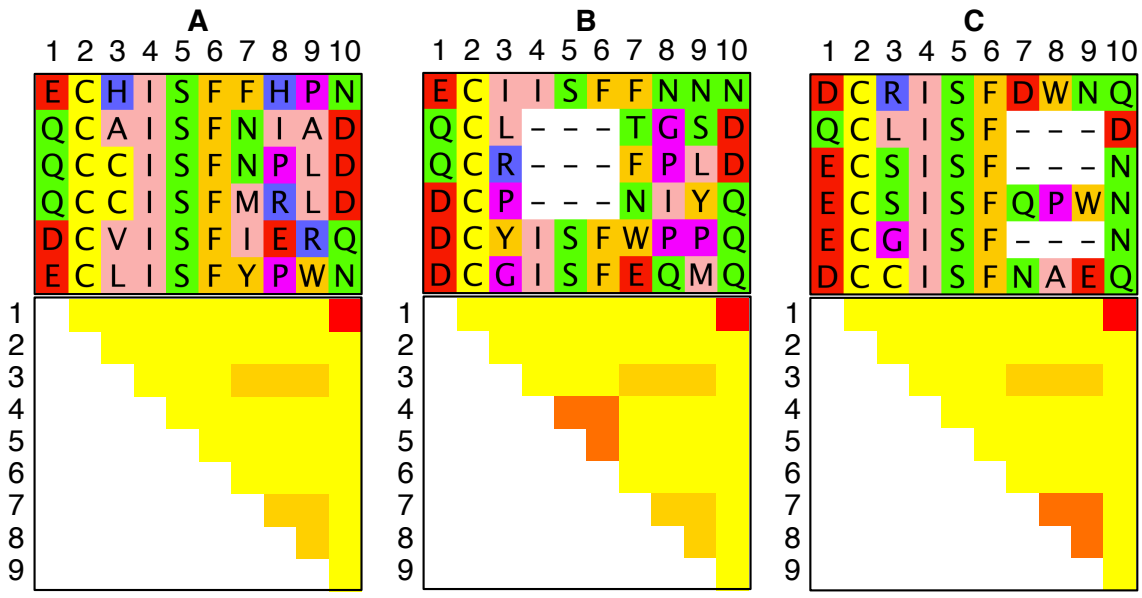

Figure 4.1: Illustrative alignments of a small hypothetical protein segment. Shown here are the first 6 rows of a 500 row multiple sequence alignment in which positions 2 and 4-6 are conserved , positions 3 and 7-9 are randomly assorting, and positions 1 and 10 covary. Below each is the Z score derived from the *MIp* measure (also called *Zp*), where yellow denotes scores less than 0.5, dark yellow denotes a score between 0.5 and 1.5, orange denotes a score between 1.5 and 2.5 and red denotes a score greater than 2.5. Gaps are shown in the alignment as the - character.

4.4 Results

4.4.1 Coevolution analysis of synthetic alignment reveals gap effects

Consider the hypothetical protein family alignments in Figure 4.1. Position 2 is conserved and does not vary within the protein family. Positions 3, 7, 8, and 9 are randomly assorting. A common assumption is that the conservation of position 2 implies it is somehow important to the protein family, as the sequence meanderings of evolution would have changed its identity if it were not. Conversely, positions exhibiting variation imply that the position is comparatively unimportant, or some evolutionary pressure would have constrained the residues allowed at that position.

Consider a simple game where a wager can be placed on the identity of an amino acid

at a position. Assume that the gambler is conservative and only wagers when confident on a correct answer. If the wager was on the identity of the amino acid at position 2 in Figure 4.1, the bettor would certainly choose cysteine. In contrast the bettor would likely choose not to wager on the identity of the amino acids at positions 3 or 7-9. In Information Theoretic terms, the uncertainty when placing your bets can be measured using Entropy (H), which is the measure of uncertainty of the identity of a random variable. Position 3 has high entropy and position 2 has low entropy.

Now consider positions 1 and 10. Here the residues can take on three possible states in each position. Without further information the bettor would be uncertain which to choose. However, what if the game is changed slightly to give the bettor one additional piece of information - the identity and location of another residue in the same protein sequence. If the bettor is gambling on the identity of position 1, neither position 2 nor position 3 provides any additional information for placing the bet. However, positions 1 and 10 are coupled: a residue at position 1 is always paired with a particular residue at position 10. So given the identity of the residue at either position, the bettor is now certain about the residue at the other. These two positions exhibit *covariation*. In the same way that the lack of variation in position 2 implies the important conservation of residue identity, the covariation between positions 1 and 10 indicates the potential for an important conserved *interaction* between the two positions.

In Information Theoretic terms, the dependency between two positions can be calculated using Mutual Information (MI). Mutual Information can be formally defined as:

$$MI_{i,j} = H_i + H_j - H_{i,j} \tag{4.1}$$

where H_i is the entropy of position i and $H_{i,j}$ is the joint entropy of positons i and j. Since high entropy translates into low certainty, equation 1 shows that MI can be understood intuitively as the reduction of uncertainty of the identity of one position when given the identity of another.

MI by itself is not generally used to estimate covariation because this measure is very sensitive to the entropy of the columns [18, 10], and because of the confounding effect of the

intrinsic phylogenetic relationships between positions in a multiple sequence alignment [8]. One commonly-used metric is MIp, which is defined as:

$$MIp_{i,j} = MI_{i,j} - \frac{\overline{MI_i}, \times \overline{MI_j},}{\overline{MI}}. \tag{4.2}$$

This measure subtracts the mean evolutionary relationships between positions, and others have developed similar measures based on regression [13]. The "heatmap" drawn below the sample alignment corresponds to the *MIp* score between positions. For example, the aforementioned covariation between columns 1 and 10 is illustrated by the red square in the heatmap, which corresponds to a normalized *MIp* score greater than 2.5.

As mentioned above, MIp-based coevolution methods have been modified frequently to analyze gapped positions in an alignment by interpreting every gap character as the 21st character in the alignment alphabet [31, 25, 14]. Essentially, these groups treat gapped positions identically to non-gapped positions. The motivation for this change seems laudable if it is assumed that gaps contain information. However, as shown below, the gambling game analogy intuitively demonstrates why MIp will produce statistical artifacts over useful information if gap characters are included as the 21st character. This demonstration shows why the proper treatment of gaps is critical to avoiding erroneous interpretations of covariation measures.

Returning to the game, now consider columns 4 through 6 in Figure 4.1A. The residues in these columns are absolutely conserved, as in column 2, and the bettor would not require any information other than this knowledge if asked to choose the identity of the residue. Note that the MIp between these columns is 0: there is no additional information that can be gained about a position by giving information about a second position since there is no uncertainty for these positions ($H = 0$).

In Figure 4.1B, a deletion event has been introduced at the conserved positions, 4 through 6 from panel A. That there is a gap at these positions does not provide new knowledge regarding the structural or functional relationships between the *ungapped* residues. This is because the

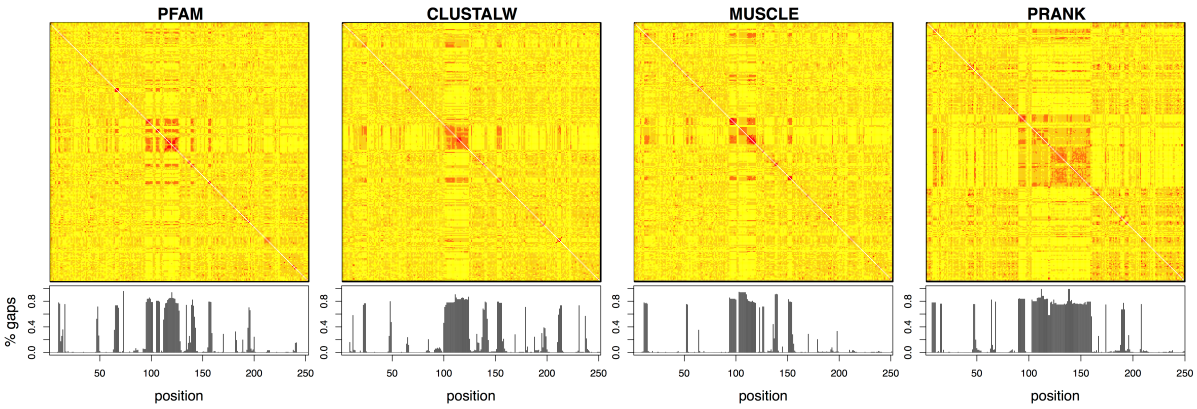

Figure 4.2: Pairwise covariation scores shown as heatmaps compared to the percentage of gaps at each position from four different alignment methods. Each node in the heatmap represents a Zp covariation score ranging from below 0.5 (yellow) to above 2.5 (red). The histogram below shows the percentage of gaps at each position corresponding to the heatmap above.

gap character formally represents a residue in one sequence matching nothing in the other [24]. However, if the gap character is considered to be an additional symbol, then entropy is added to the positions and information is created. When *MIp* is calculated, we observe a dramatic increase in the covariation values *that is local to the deletion event itself*. An investigator might therefore conclude that these positions coevolve strongly. This inference would be wrong because the only information we have added is that, at these positions, nothing in one sequence corresponds to nothing in another. As discussed later, the information added to the gap position is dependent on the method used to align the sequences and on the assumptions used by that method.

In Figure 4.1C, a deletion event has occurred at the randomly-assorting positions, 7 through 9. Again, there is a dramatic increase in local covariation that is local to the gapped positions. Note however, that since the non-gapped residues are randomly assorting, the residues that are not in gaps are still assorting completely randomly: thus knowing the identity of a residue at one position continues to provide no information about the identity of a residue at a different position.

The simple examples discussed in Figure 4.1 show that *arbitrarily placed gap characters*

invariably result in the gain of information. This results in local covariation being increased dramatically. However, the gap placement rules used by the multiple sequence alignment programs in widespread use are different, some are arbitrary some are not. It is thus possible that gap placement by at least some of these automated algorithms does not create information, and as as result does not increase local covariation scores.

4.4.2 Gap positions show high covariation in real alignments

We examined the alignments generated by four major multiple sequence alignment methods to address this question. One alignment was collected from PFAM, which uses a method where a Hidden Markov Model of the protein family is built from curated, high-quality alignments, and extended to embrace new members of the family as they are discovered [21]. The same sequences from the PFAM alignment were realigned using three other methods. One is progressive multiple sequence alignment as used by ClustalW [27]. Here gap placement is chosen using an affine gap score that is dynamically weighted to include both the local amino acid composition and the location of gaps in alignments between highly similar sequences. More recent approaches include iterative refinement of the alignment to optimize a scoring function, as used in MUSCLE [9], or to optimize gap placement based on the inferred phylogenetic tree, as used in PRANK [15].

Figure 4.2 shows the correspondence between gapped positions and high covariation scores in real alignments when gap characters are treated as the 21st amino acid in coevolution analysis. Pairwise covariation scores are shown in a large heatmap with the same colour scheme as figure 4.1, and a histogram of corresponding prevalence of gaps at each position is provided below. Here it is demonstrated that the highest-scoring covarying pairs (red) are almost exclusively related by being in the same contiguous gapped segment. Similar results are found regardless of the alignment method, demonstrating that the phenomenon is not restricted to a single alignment method. Rather, the effect appears to be universal across 4 different alignment methods which represent 4 different alignment strategies.

Alignment algorithms often attempt to minimize the occurrence of gaps and maximize residue identity which can lead to high covariation within gapped regions. Multiple independent insertions or deletion events may therefore be aligned together as a single event represented by a single column, thus avoiding multiple gap initiation penalties. Essentially, these gapped positions appear optimized according to the criteria of the method, but may be misleading according to phylogenetic or coevolutionary inference [29]. Though gapped positions and the associated surrounding residues may contain the evolutionary information if they are assigned correctly [23], they are frequently rejected altogether in evolutionary analysis because the accurate gap placement may be too difficult to assign [3, 26]. The PRANK phylogeny-aware strategy attempts to align only homologous insertions and deletions, which means that it is attempting to insert gaps into the alignment with more biological relevance, yet it also shows a strong correspondence between gapped positions and covariation score (Figure 4.2).

Finally, we note that it is apparent that high covariation scores often exist between gapped segments. These can be easily observed in the four panels above as off-diagonal patches of high mutual information centred on the gapped regions. Contact maps derived from such analyses would lead to the inference that all gapped positions are in clustered together in contact in the corresponding structure.

4.4.3 Database wide screen demonstrates low contact prediction accuracy in gapped positions

To demonstrate the erroneous predictions that covariation statistics make when dealing with gapped positions, we analyzed the contact prediction accuracy of *MIp* across all suitable alignments from the Conserved Domain Database (CDD) [17], which are curated with the Cn3D tool. Figure 4.3 shows that the prediction accuracy of *MIp* is much poorer when analyzing gap positions than when analyzing positions with no gap characters. A modified version of the standard MIpToolset [5] covariation software was run across a dataset of 314 CDD alignments to generate *MIp* scores. *MIp* scores normally predict inter-residue contact with high accuracy in

exclusively non-gapped positions. However, the modified version of the MIpToolset allowed gapped positions to be included in the analysis. Contact prediction is commonly used as a proxy for coevolution when benchmarking covariation statistics like *MIp* because there is no way to objectively obtain the set of truly coevolving residues from experiment. Only pairs of positions that are more than 6 positions apart in sequence were included in the analysis because residues which are close in sequence are trivially in contact.

In figure 4.3, the pairs of positions containing no gap characters (filled squares) show strong contact prediction accuracy, nearly 80% of the top-scoring pairs in the 314 alignments found to be in contact. Conversely, *MIp* is not able to reliably identify contacting pairs of positions within the same gap (open circles), with the mean fraction of pairs in contact never reaching 20% accuracy. This observation of the effects of database wide differences between gapped and untapped positions supports the previous examples.The low contact prediction accuracy of gapped positions implies that these predicted coevolving pairs are false positives.

4.5 Discussion

Gap placement and covariation statistic calculation is problematic for the automated alignment tools shown above, but perhaps it is possible that manual placement of gaps by an expert annotator can overcome these problems. We have noted previously however [6, 7], that the problem of high local covariation scores within gaps persists even in structurally-supported alignments such as the Conserved Domain Database (CDD) [17]. The CDD database is built using Cn3D, an excellent curation tool for integrating structural and sequential information when creating a new protein family alignment. However, using Cn3D (or alignments generated from Cn3D) requires an understanding of the assumptions the tool uses. The Cn3D documentation states, "*that the unaligned residues are displayed as a convenience to the user, but nothing can nor should be inferred from the apparent 'alignment' of residues in the unaligned areas*" [4]. So little information is contained in gapped segments that Cn3D explicitly permits the user to

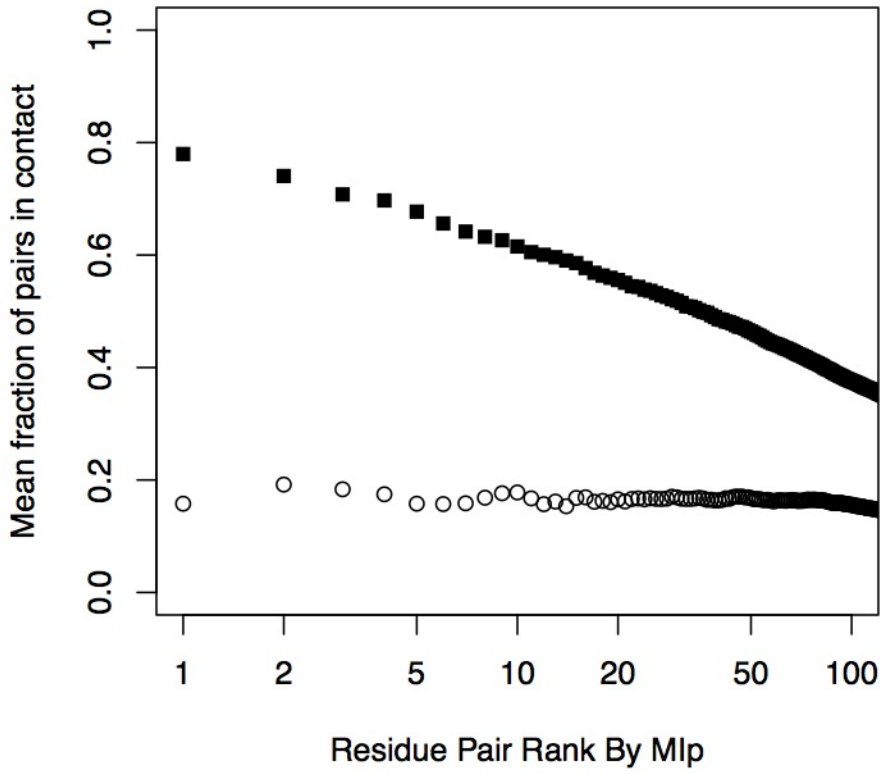

Figure 4.3: Contact prediction accuracy by *MIp* of pairs with no gap characters (filled squares) and pairs within a single contiguous gap (open circles). *MIp* scores were generated for 314 CDD alignments. These pairs were ranked from highest to lowest scoring. The mean fraction of pairs in contact for the top 1 to 100 pairs across the 314 alignments are shown.

alter the location of residues positioned in the gapped regions. Similar issues result [7] with alignments from the highly curated BALiBASE dataset [28].

Thus, the increased covariation scores between positions contained in the same gapped segment should be viewed as nothing more than a statistical artifact. Insertions and deletions typically occur in unstructured, surface-exposed loops, so including gaps as characters in a co-evolutionary analysis will likely lead to a misleading conclusion that such regions structurally or functionally important. Mutual Information-based statistics such as *MIr* [18], *MIp* (and its normalized form *Zp*) [8], *Zpx* [6], ΔZp [7], and *Zres* [13], should not interpret a gap character as an amino acid, unless rigorous analysis has shown that a particular method is able to overcome the bias towards enhancing the score between gapped residues.

The treatment of gaps in evolutionary analysis has long been controversial [23], notably following the assertion that identical indels can be homoplasious and thus misleading [11]. As demonstrated, including gaps as the 21st amino acid leads to a marked increase in covariation score (Figure 4.1), which leads to an overrepresentation of gapped positions as most highly covarying pairs (Figure 4.2), and finally causes a notable decrease in the prediction of putatively coevolving pairs by the standard benchmark (Figure 4.3). Both the decrease in accuracy and the over-emphasis of gap regions when misusing Mutual Information-based statistics are important. For example, Liu and Bahar suggested a correlation exists between coevolving pairs of positions and structural dynamics, but their analysis included gaps as the 21st character [14]. The fact that gaps tend to occur in flexible, unstructured surface loops suggests that the inclusion of gap characters may have affected their conclusions. Another example is Marcos et al., who benchmarked their Direct-Coupling Analysis coevolution method against non-corrected Mutual Information and included gaps as the 21st character in their *MI* calculation [19]. The low performance of Mutual Information in their comparison is likely due to the low accuracy of predictions in gapped positions (Figure 4.3), and the high scores caused by including the gap character (Figure 4.2).

We believe that the coevolution analysis of gaps as the 21st amino acid is unequivocally

incorrect. Support for this assertion comes directly from Figure 4.2 where it can be seen that the inclusion of a gap, into either conserved or non-conserved columns, automatically increases the apparent information in the columns **without an increase in actual information**. Thus, we recommend caution when drawing any conclusions from any gapped position in an alignment. As mentioned above, many alignment strategies introduce gaps to sequences in a manner that is inappropriate for coevolution analysis. There are various strategies employed by alignment algorithms that decide where gaps should be placed; these strategies generate alignment within gap regions of vastly different quality.

Another major challenge in coevolution analysis is validating the predictions of a given coevolution method. Coevolution prediction shares the same inferential challenges of phylogeny building, but lacks the luxury of using multiple positions as "independent" evidence. Structural contact is typically used as a proxy for true-positives in coevolutionary analysis, as it is hypothesized that residues must be close enough to interact in order to coevolve; this hypothesis is contradicted and complicated by indirect coevolution through an intermediate residue [2].

Contact is only meaningful if the two positions in question are not trivially in contact because of sequence proximity. However, two positions that are distant in sequence but have contacting side chains represent an unlikely and potentially important interaction. It is imperative to analyze the structures in a given dataset to determine an adequate "sequence-distance" cutoff when analyzing coevolution predictions. In this analysis a sequence-distance cutoff of 6 was used in order to exclude fewer intra-gap pairs; however, in typical benchmarking scenarios we recommend using a stricter cutoff such as 10.

4.6 Conclusion

In conclusion, we wish to recommend caution when analyzing positions that contain gaps using coevolutionary tools. Mutual Information-based algorithms, including but not limited to *MIp* and its normalized form *Zp*, *Zpx*, ΔZp, *Zres*, should not be modified to include gap characters

unless future modifications are shown to rigorously compensate for the errors this introduces since none of these measures were designed to analyze gaps.

Bibliography

[1] W R Atchley, K R Wollenberg, W M Fitch, W Terhalle, and A W Dress. Correlations among amino acid sites in bHLH protein domains: an information theoretic analysis. *Mol Biol Evol*, 17(1):164–178, January 2000.

[2] Lukas Burger and Erik van Nimwegen. Disentangling direct from indirect co-evolution of residues in protein alignments. *PLoS Computational Biology*, 6(1):e1000633, January 2010.

[3] J Castresana. Selection of conserved blocks from multiple alignments for their use in phylogenetic analysis. *Molecular Biology and Evolution*, 17(4):540, 2000.

[4] Cn3d Tutorial. Structure Alignments in Cn3D. September 2011.

[5] R J Dickson and G B Gloor. The MIp Toolset: an efficient algorithm for calculating Mutual Information in protein alignments. *arXiv.org*, 2013.

[6] RJ Dickson, LM Wahl, AD Fernandes, and GB Gloor. Identifying and seeing beyond multiple sequence alignment errors using intra-molecular protein covariation. *PLoS ONE*, 5(6):e11082, 2010.

[7] Russell J Dickson and Gregory B Gloor. Protein sequence alignment analysis by local covariation: coevolution statistics detect benchmark alignment errors. *PLoS ONE*, 7(6):e37645, 2012.

[8] S D Dunn, L M Wahl, and G B Gloor. Mutual information without the influence of phylogeny or entropy dramatically improves residue contact prediction. *Bioinformatics*, 24(3):333–340, January 2008.

[9] Robert C Edgar. MUSCLE: a multiple sequence alignment method with reduced time and space complexity. *BMC Bioinformatics*, 5:113, August 2004.

[10] Gregory B Gloor, Louise C Martin, Lindi M Wahl, and Stanley D Dunn. Mutual information in protein multiple sequence alignments reveals two classes of coevolving positions. *Biochemistry*, 44(19):7156–7165, May 2005.

[11] E M Golenberg, M T Clegg, M L Durbin, J Doebley, and D P Ma. Evolution of a noncoding region of the chloroplast genome. *Molecular phylogenetics and evolution*, 2(1):52–64, March 1993.

[12] C W Hogue. Cn3D: a new generation of three-dimensional molecular structure viewer. *Trends Biochem Sci*, 22(8):314–316, August 1997.

[13] Daniel Y Little and Lu Chen. Identification of Coevolving Residues and Coevolution Potentials Emphasizing Structure, Bond Formation and Catalytic Coordination in Protein Evolution. *PLoS ONE*, 4(3):e4762, March 2009.

[14] Ying Liu and Ivet Bahar. Sequence Evolution Correlates with Structural Dynamics. *Molecular Biology and Evolution*, April 2012.

[15] A Loytynoja and N Goldman. Phylogeny-Aware Gap Placement Prevents Errors in Sequence Alignment and Evolutionary Analysis. *Science*, 320(5883):1632–1635, June 2008.

[16] Ari Löytynoja and Nick Goldman. webPRANK: a phylogeny-aware multiple sequence aligner with interactive alignment browser. *BMC Bioinformatics*, 11:579, 2010.

[17] Aron Marchler-Bauer, John B Anderson, Farideh Chitsaz, Myra K Derbyshire, Carol DeWeese-Scott, Jessica H Fong, Lewis Y Geer, Renata C Geer, Noreen R Gonzales, Marc Gwadz, Siqian He, David I Hurwitz, John D Jackson, Zhaoxi Ke, Christopher J Lanczycki, Cynthia A Liebert, Chunlei Liu, Fu Lu, Shennan Lu, Gabriele H Marchler, Mikhail Mullokandov, James S Song, Asba Tasneem, Narmada Thanki, Roxanne A Yamashita, Dachuan Zhang, Naigong Zhang, and Stephen H Bryant. CDD: specific functional annotation with the Conserved Domain Database. *Nucleic Acids Res*, 37(Database issue):D205–10, 2009.

[18] L C Martin, G B Gloor, S D Dunn, and L M Wahl. Using information theory to search for co-evolving residues in proteins. *Bioinformatics*, 21(22):4116–4124, 2005.

[19] Faruck Morcos, Andrea Pagnani, Bryan Lunt, Arianna Bertolino, Debora S Marks, Chris Sander, Riccardo Zecchina, José N Onuchic, Terence Hwa, and Martin Weigt. Direct-coupling analysis of residue coevolution captures native contacts across many protein families. *Proceedings of the National Academy of Sciences*, 108(49):E1293–301, December 2011.

[20] C Notredame, DG Higgins, and J Heringa. T-coffee: a novel method for fast and accurate multiple sequence alignment1. *Journal of Molecular Biology*, 302(1):205–217, 2000.

[21] Marco Punta, Penny C Coggill, Ruth Y Eberhardt, Jaina Mistry, John Tate, Chris Boursnell, Ningze Pang, Kristoffer Forslund, Goran Ceric, Jody Clements, Andreas Heger, Liisa Holm, Erik L L Sonnhammer, Sean R Eddy, Alex Bateman, and Robert D Finn. The Pfam protein families database. *Nucleic Acids Res*, 40(Database issue):D290–301, January 2012.

[22] R Core Team. R: A Language and Environment for Statistical Computing. 2013.

[23] M P Simmons and H Ochoterena. Gaps as characters in sequence-based phylogenetic analyses. *Systematic biology*, 49(2):369–381, June 2000.

[24] T F Smith and M S Waterman. Identification of common molecular subsequences. *Journal of Molecular Biology*, 147(1):195–197, March 1981.

[25] Janardanan Sreekumar, Cajo J F ter Braak, Roeland C H J van Ham, and Aalt D J van Dijk. Correlated mutations via regularized multinomial regression. *BMC Bioinformatics*, 12:444, 2011.

[26] G Talavera and J Castresana. Improvement of phylogenies after removing divergent and ambiguously aligned blocks from protein sequence alignments. *Systematic biology*, 56(4):564, 2007.

[27] J D Thompson, D G Higgins, and T J Gibson. CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Res*, 22(22):4673–4680, November 1994.

[28] Julie D Thompson, Patrice Koehl, Raymond Ripp, and Olivier Poch. BAliBASE 3.0: latest developments of the multiple sequence alignment benchmark. *Proteins: Structure, Function, and Bioinformatics*, 61(1):127–136, October 2005.

[29] Tandy Warnow. Standard maximum likelihood analyses of alignments with gaps can be statistically inconsistent. *PLoS currents*, 4:RRN1308, 2012.

[30] Andrew M Waterhouse, James B Procter, David M A Martin, Michèle Clamp, and Geoffrey J Barton. Jalview Version 2–a multiple sequence alignment editor and analysis workbench. *Bioinformatics*, 25(9):1189–1191, May 2009.

[31] Kevin Y Yip, Prianka Patel, Philip M Kim, Donald M Engelman, Drew McDermott, and Mark Gerstein. An integrated system for studying residue coevolution in proteins. *Bioinformatics*, 24(2):290–292, January 2008.

Chapter 5

Multiple sequence alignment errors and coevolution

5.1 Abstract

5.1.1 Background

There is currently no way to verify the quality of a multiple sequence alignment that is independent of the assumptions used to build it. Sequence alignments are typically evaluated by a number of established criteria: sequence conservation, the number of aligned residues, the frequency of gaps, and the probable correct gap placement. Covariation analysis is used to find putatively important residue pairs in a sequence alignment. Different alignments of the same protein family give different results demonstrating that covariation depends on the quality of the sequence alignment. We thus hypothesized that current criteria are insufficient to build alignments for use with covariation analyses.

A version of this chapter has been published.
RJ Dickson, LM Wahl, AD Fernandes, and GB Gloor. (2010). Identifying and Seeing beyond Multiple Sequence Alignment Errors Using Intra-Molecular Protein Covariation. PLoS ONE.5(6): e11082

5.1.2 Methodology/Principal Findings

We show that current criteria are insufficient to build alignments for use with covariation analyses as systematic sequence alignment errors are present even in hand-curated structure-based alignment datasets like those from the Conserved Domain Database. We show that current non-parametric covariation statistics are sensitive to sequence misalignments and that this sensitivity can be used to identify systematic alignment errors. We demonstrate that removing alignment errors due to 1) improper structure alignment, 2) the presence of paralogous sequences, and 3) partial or otherwise erroneous sequences, improves contact prediction by covariation analysis. Finally we describe two non-parametric covariation statistics that are less sensitive to sequence alignment errors than those described previously in the literature.

5.1.3 Conclusions/Significance

Protein alignments with errors lead to false positive and false negative conclusions (incorrect assignment of covariation and conservation, respectively). Covariation analysis can provide a verification step, independent of traditional criteria, to identify systematic misalignments in protein alignments. Two non-parametric statistics are shown to be somewhat insensitive to misalignment errors, providing increased confidence in contact prediction when analyzing alignments with erroneous regions because of an emphasis on they emphasize pairwise covariation over group covariation.

5.2 Introduction

Two or more variable positions in a protein may coadapt to conserve interactions needed for proper structure or function [6, 30, 21]. Strong covariation between pairs of positions is taken to indicate the presence of coadaptation events which are maintained in the alignment as coevolution. It is often assumed that coadapted residues contact each other in the folded protein structure [6, 26, 8, 5], thus the proportion of putative coadapted positions in contact is often

used to benchmark covariation methods. This assumption is not bidirectional, only a small proportion of contacting sites are thought to coevolve strongly [6, 21]. Furthermore, pairs that are very close in sequence are trivially in contact and thus are disregarded when evaluating covariation statistics.

Covariation statistics are often used to aid in residue contact identification, *de novo* protein structure prediction and structure-function analysis [22, 16, 24, 23, 27]. Indeed, the first step in many structure prediction algorithms is a multiple sequence alignment followed by some sort of covariation measure. Even predictions with modest accuracies are helpful because they restrict the positions to be examined. Standard benchmarks for covariation accuracy measure the fraction of covarying amino acid pairs that are in contact. There are a large number of methods to identify covarying positions [31] in part because some methods work better on certain alignments than on others. Many groups have observed high covariation between residues close in sequence—leading to the belief that two or more positions can only be identified as coevolving if they are some minimum distance apart in sequence. However, little attention is paid to the overall problem that there is no consensus as to the characteristics of truly covarying positions. This results in a counter-intuitive situation where contact is used as a proxy for covariation for benchmarking purposes, but traditional measures like sensitivity and specificity of contact prediction are not very meaningful because only some contacting pairs covary.

Atchley et al. [1, 28] suggested that covariation observed between positions i and j in a protein is composed of a signal from 1) structural or 2) functional constraints, 3) background noise contributed by shared phylogenetic ancestry, and 4) stochastic events. Thus, the structural and functional signal is superimposed on the background noise contributed by phylogeny and by random processes. As implied by this model there are several intrinsic limitations to detecting coevolution between amino acid positions in protein families. First, the sequence alignments must contain sufficient sequences with enough variation for the signal to exceed the noise. Estimates of the required number of sequences needed in the alignments for this to be true vary from ~30 [5] to >125 [16, 26, 19, 2]. Secondly, all positions in a protein ap-

pear to covary because of their shared ancestry, and this signal is the only systematic source of covariation for the vast majority of position pairs [5, 28, 3]. We recently showed that the phylogenetic signal was similar for all positions in a protein family and that it could be estimated as the product of the average covariation of positions i and j with all other positions [3]. This resulted in 'product-corrected Mutual Information' MIp and its transform, Zp, which was much more sensitive and specific than other previous non-parametric methods.

The goal of multiple sequence alignment is to place residues from each sequence in the protein family at homologous positions in the final sequence alignment. The multiple sequence alignment for a given protein family is usually different in the various standard collections of protein families and the disagreement between protein datasets demonstrates that all multiple sequence alignment methods produce errors. Furthermore, the placement of a gap in the protein family is an explicit acknowledgement that no homologous position exists for one or more members of the protein family. The difficulty in generating a sequence alignment is highlighted by the observation that even structure-based alignment methods disagree [12, 14]. As one example, structure-based alignment algorithms are susceptible to shift error [12], meaning that positions in the structure alignment are not orthologous despite the fact that much of the secondary structural elements seem to overlap between aligned structures.

We observed that the same protein family often gave different numbers of covarying positions when alignments were from different sources even if the alignments contained comparable numbers of sequences. We also found that alignments generated without structural information identified fewer pairs in contact in the folded protein compared to alignments generated with structural information. These observations suggested that the quality of the alignment made a large contribution to the background covariation signal. This observation is supported by Wong et al. [29] who found that alignments of the same sequence dataset by different methods lead to different conclusions in comparative genomics studies.

Here we examine the effect of systematic misalignment on covariation scores. We demonstrate that alignment errors lead to incorrect conclusions about covariation and conservation.

We show that Zp can be used to identify systematic misalignments in protein families. Furthermore, we show that new statistics ΔZp and Zpx are relatively insensitive to systematic alignment errors, and are especially effective at identifying pairwise covarying residues. Significantly, these two corrections identify substantially different populations of covarying pairs with similar accuracy.

5.3 Results

5.3.1 Systematic sources of error

Many commonly-used multiple sequence alignment programs use the progressive sequence alignment strategy in which the alignment and the locations of insertions and deletions are permanently fixed in the growing alignment [4]. An alternative method is structure-based multiple sequence alignment which aligns three-dimensional protein structures and then aligns sequences progressively to the initial structure alignment [11]. Both structure-based and progressive methods systematically propagate early errors through the alignment. Another alignment strategy is iterative alignment, where progressive alignments are built and then iteratively refined to attempt to remove errors that are introduced in the growing progressive alignment [10]. The phylogeny-aware strategy attempts to minimize systematic errors by preventing incorrect gap placement [17]. While it is clear that each alignment strategy is susceptible to varying types and amounts of systematic error, we began by approximating it by using a simple experiment to estimate its effects.

The impact of systematic errors on the estimation of covariation was tested in an alignment of triosephosphate isomerase by randomly shifting a fraction of the sequences in one aligned segment left or right by 1 residue. We chose to shift between 0 and 30% of the sequences within the selected segment positions—a range chosen because the commonly-used multiple sequence alignment programs have between 70% and 80% accuracy [4]. Figure 5.1 shows that the Zp signal increased if both positions in a pair were in the misaligned segment (red)

and decreased if one of the positions in the pair was in the misaligned segment and the other was outside the segment (green) when compared with the pairs unaffected by the misalignment (blue).

The increased *Zp* values were not distributed evenly among all positions in the alignment (Fig. 5.1, 5.2). Figure 5.1 shows that misaligned pairs have a marked increase in *Zp* score with other residues in the misaligned region. There are several interesting features of such misaligned families: 1) Contacting pairs of positions in the properly aligned regions tend to be assigned *Zp* scores that are much higher than the mean, but often not large enough to stand out against the misaligned region. However, there is often a large difference between the score of a contacting pair and the next-highest score. 2) Noncontacting pairs tend to have small differences between consecutive scores. 3) High-scoring pairs due to misalignment tend to cluster together. Thus we introduce ΔZp, a relative *Zp* measure (Materials and Methods), to compensate for high-scoring noncontacting pairs due to misalignment.

Figure 5.2 shows a plot of the mean *Zp* score between all pairs of positions in a 6-residue window for a structure-guided alignment of triosephosphate isomerase. The first third of the misaligned segment was highly conserved and had very low *Zp* scores, the remainder was highly entropic and had higher initial *Zp* scores. The systematic misalignment of even 5% of the sequences resulted in a dramatic increase in local *Zp* values for the conserved but not for the non-conserved portion of the segment (Fig. 5.2). This effect was even more pronounced when larger fractions of the multiple sequence alignment were misaligned. We concluded that systematic sequence misalignment resulted in a characteristic pattern of elevated local *Zp* scores which was most extreme for conserved segments.

The effect of systematic misalignments on the underlying information theoretic values is shown in Figure 5.3. We consider an arbitrary four residue sequence where each position is conserved (panel A). Each position in this alignment contains no entropy meaning that the residue at that position is certain. Because there is no variation, there cannot be any covariation and thus the values for any covariation statistic is 0. However, when half of the sequences are

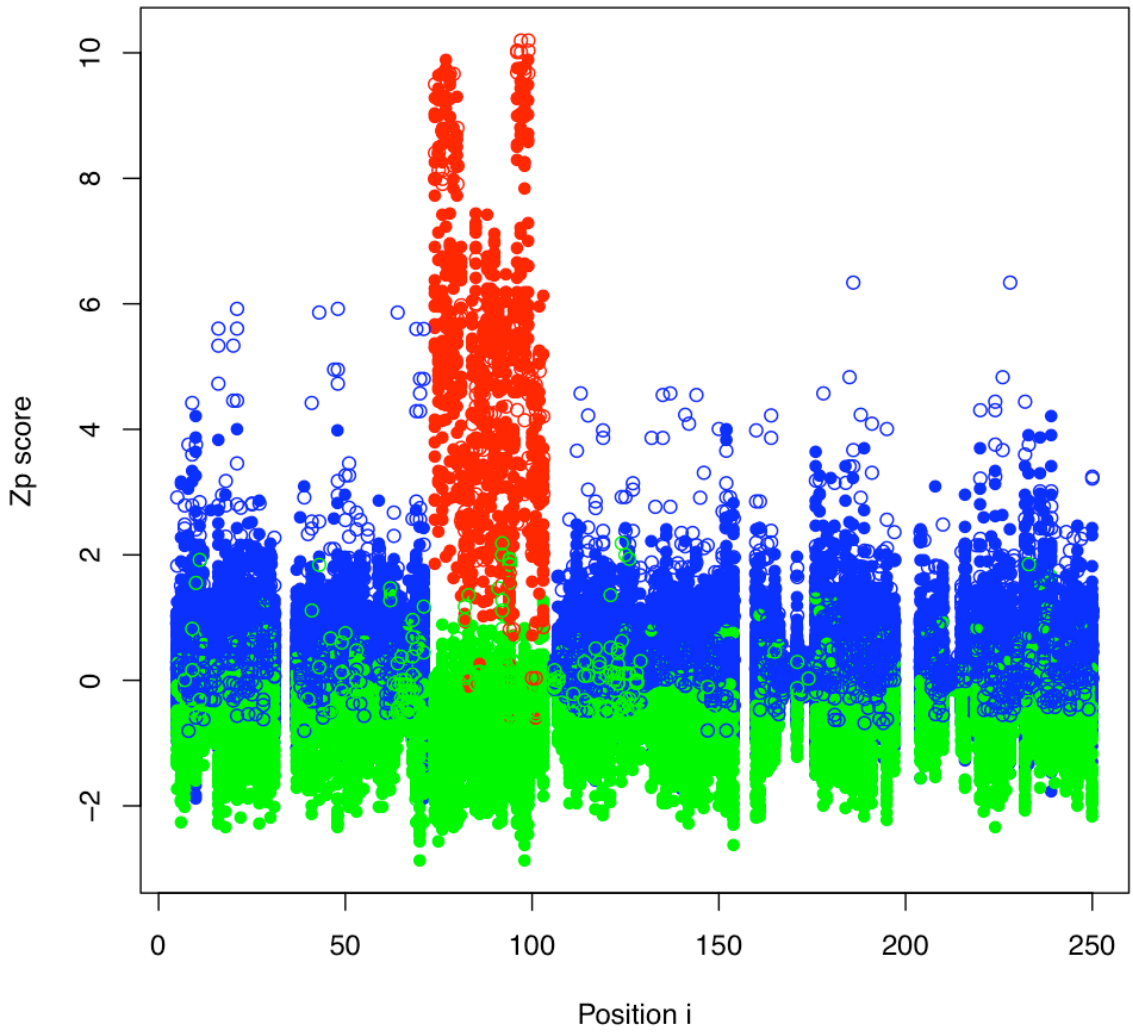

Figure 5.1: Misalignments cause increased covariation scores. All pairwise *Zp* scores are shown by position in a triosephosphate isomerase alignment which contains a synthetic systematic misalignment in the 4th ungapped segment. Pairs where both positions are from within the misaligned segment are shown in red. Positions where both positions are from outside the misaligned segment are shown in blue. Positions where one position is within and one position is outside the misaligned segment are shown in green. Positions in contact are represented by white-filled circles, positions not in contact are represented by solid colours. The misalignment was made by shifting 20% of sequences in the red-highlighted region by one position to the left or right. Intra-misalignment pairs (red) have higher *Zp* scores and inter-misalignment pairs (green) have lower *Zp* scores when compared to the normal pair distribution (blue).

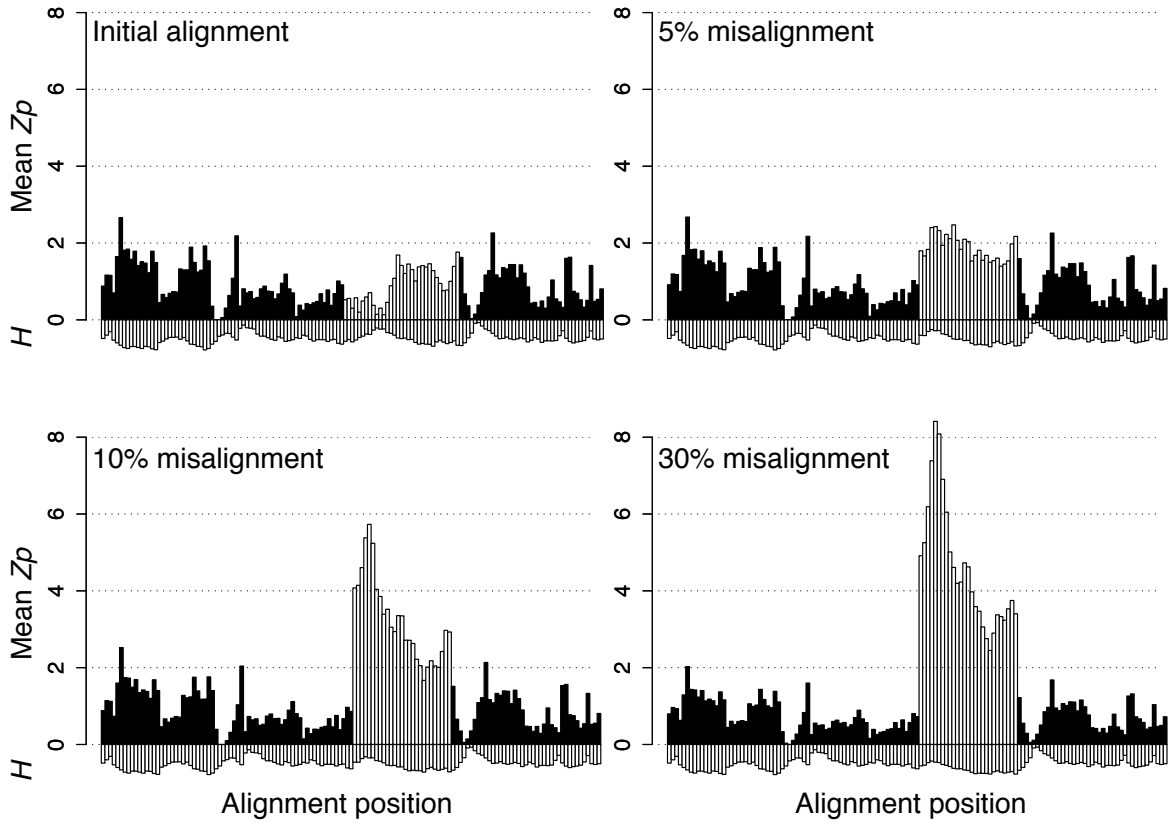

Figure 5.2: Systematically misaligned regions have high local *Zp* values. The plots show the mean *Zp* score for all pairs of positions in overlapping 6-residue windows versus the window start position. The light bars show the segment of the alignment that was systematically misaligned with the fraction of misaligned sequences indicated. The mean entropy (*H*) of the positions in the same window multiplied by -1 is shown below.

shifted to the right by one residue, the entropy and joint entropy of the positions increase. The reason that positions 2, 3, and 4 covary is easy to understand intuitively: if you are given the alignment and the identity of a residue at one position, you know the identity of the other two with 100% confidence. It is tempting to think that this effect is simply due to the increase in entropy at each position; this is demonstrably untrue as the effect is still visible when using the covariation statistic *Zp*, which does not correlate with entropy [3].

To demonstrate the increase in *Zp*, the 5-position misaligned block in Figure 5.3 was attached to the N-terminus of a triosephosphate isomerase alignment so *Zp* could be estimated. The covariation in the misaligned region can not be due to shared ancestry, but rather is an

A. Conserved Positions

1	2	3	4	5
C	D	E	F	–
C	D	E	F	–
C	D	E	F	–
C	D	E	F	–

B. Conserved H

1	2	3	4
0	0	0	0

C. Conserved Joint$H$

	1	2	3	4
1	–	0	0	0
2	0	–	0	0
3	0	0	–	0
4	0	0	0	–

D. Conserved MI

	1	2	3	4
1	–	0	0	0
2	0	–	0	0
3	0	0	–	0
4	0	0	0	–

E. Conserved Zp

	1	2	3	4
1	–	-0.01	-0.01	-0.01
2	-0.01	–	-0.01	-0.01
3	-0.01	-0.01	–	-0.01
4	-0.01	-0.01	-0.01	–

F. Shifted Positions

1	2	3	4	5
C	D	E	F	–
–	C	D	E	F
C	D	E	F	–
–	C	D	E	F

G. Shifted H

2	3	4
0.23	0.23	0.23

H. Shifted Joint$H$

	2	3	4
2	–	0.23	0.23
3	0.23	–	0.23
4	0.23	0.23	–

I. Shifted MI

	2	3	4
2	–	0.23	0.23
3	0.23	–	0.23
4	0.23	0.23	–

J. Shifted Zp

	2	3	4
1	–	20.47	20.47
2	20.47	–	20.47
3	20.47	20.47	–

Figure 5.3: Positions with shift error have high markedly increased covariation scores. When positions 1, 2, 3, and 4 are aligned correctly (A) the positions are conserved and thus there is no entropy (B), joint entropy (C) and therefore no mutual information (D) between the positions. *Zp* was calculated by replicating and attaching the sequences in (A) and (D) to the N-terminus of a triosephosphate isomerase alignment such that every sequence began with the four-residue insertion and gap at the N terminus. The *Zp* scores of the conserved positions fell below the significance threshold of 4.5 (E). The alignment was altered to simulate worst-case shift error; half the sequences were shifted one position to the right (F). These positions have the highest possible entropy (G), joint entropy (H), and mutual information (I) scores for a position with only two residues. As with the conserved alignment, the shifted alignment was inserted at the N-terminus of an alignment of triosephosphate isomerase, such that half the sequences contained a gap after the four residues and the other half contained a gap before the four residues. The resulting *Zp* scores are well above the threshold of 4.5 (J).

entirely synthetic side-effect of the alignment process and thus the phylogeny correction of *Zp* does not compensate for it. The result is a *Zp* score much higher than the 4.5 cutoff judged to be significant [3]. This effect is analogous to any position which contains information, but with a moderately decreased effect. The increase in covariation is, in fact, due to the proportionally larger increase in positional entropy to joint entropy which is localized only to the misaligned positions.

Little and Chen [15] showed that the covariation statistic, $Zres$, was capable of high accuracy when predicting contacting pairs of positions. $Zres$ emphasizes pairs of positions which covary strongly relative to the covariation distribution of positions involved, rather than the entire *Zp* distribution. We investigated whether ΔZp and $Zres$ predict contacts with high accuracy because of insensitivity to misalignment. However, we used *Zpx*, a variation on $Zres$ which is calculated more efficiently but is virtually identical (Supplementary Methods S1).

Figure 5.4 shows the difference across a 6-residue window in mean *Zp*, ΔZp, and *Zpx* (calculated as in Materials and Methods) values between the initial alignment and an alignment where an interior segment was systematically misaligned by 30%. Here, positions in the alignment were found to have mean local *Zp* scores that were up to 60-fold greater than the mean local value for the initial alignment (Figure 5.2 - Initial alignment vs. 30% misalignment). In contrast, the difference scores in ΔZp and *Zpx* were much smaller in the misaligned region. Thus, we predicted that high local *Zp* values may be useful to detect misaligned segments in protein families and that ΔZp and *Zpx* may be insensitive to misaligned segments.

5.3.2 Systematic misalignments in CDD

The above analysis on modeled data suggests that if systematically misaligned protein families exist in popular datasets, they might have segments that display high local *Zp* values. The Conserved Domain Database (CDD) [18] was examined for protein families that contained aligned segments displaying elevated mean *Zp* values in 6-residue windows; a number were identified that had 5 or more 6-residue windows with mean local *Zp* scores ≥ 2.5 (Table S1).

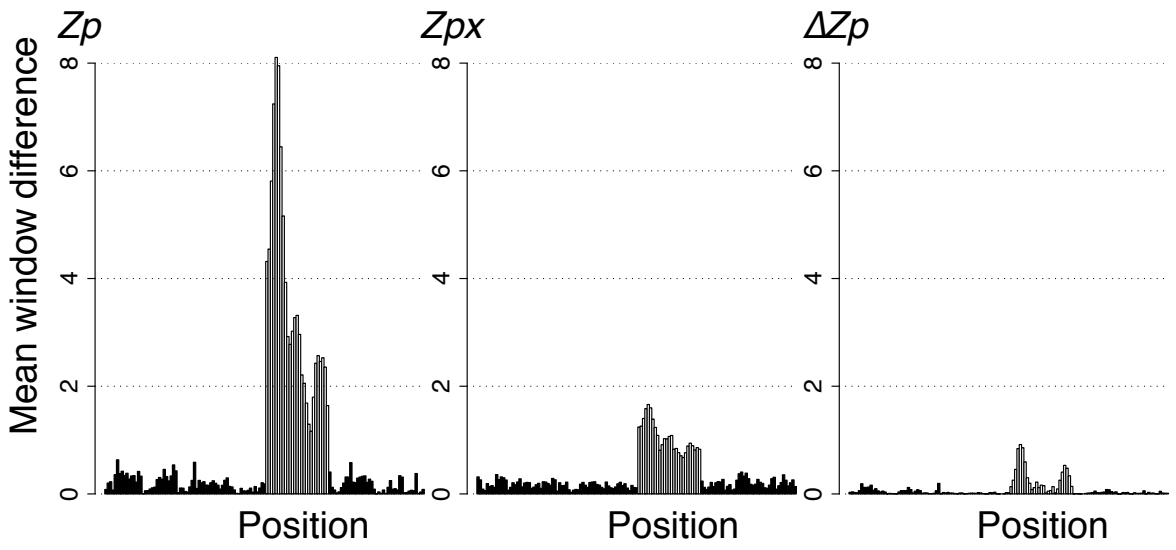

Figure 5.4: ΔZp and *Zpx* are less affected by sequence misalignments than *Zp*. The difference between the mean block scores (*Zp*, *Zpx*, ΔZp) for the reference alignment and the alignment containing 30% misalignment is plotted for each measure.

cd00300, the alignment for L-lactate dehydrogenases which is shown in Figure S1A, is one example of an alignment identified to contain systematic misalignments. Examination of the alignment found two sub-populations of sequences that did not fit the overall alignment consensus in these regions (Table S2). One population included sequences that were misaligned in the central portion of the alignment; these were found to be malate dehydrogenase sequences. The second population was composed of partial sequences that were stretched to fit the overall alignment model. Furthermore, the ungapped segments of the structure alignment were placed incorrectly. Removal of both classes of sequence as well as correcting the structure alignment resulted in a more uniform mean local *Zp* as shown in Figure S1B. Interestingly, the residues near the central catalytic core region were much more conserved when the malate dehydrogenase sequences were removed from the alignment. As expected from the modeled data, the contamination of the initial alignment by the paralogous malate dehydrogenase protein family increases *Zp* scores at the conserved active site.

Systematic misalignment errors have a dramatic effect on the predictions of a covariance method. The set of predicted covarying pairs are often visualized as a contact map, a two

dimensional array where the secondary and tertiary structure of the protein are visible [7]. Because Zp is sensitive to systematic misalignment errors, the contact map produced from the original cd00300 alignment contains predictions centered around the sites of misalignment and contains no useful structural information as shown in Figure 5.5-Zp:original. In contrast, ΔZp and Zpx produce more informative contact maps than Zp even with systematically misaligned data; however, the contact maps are largely composed of local contacts covering the secondary structure of the protein (Fig. 5.5-original). When the repaired alignment was used, the contact maps of all three measures show predictions across the length of the protein encompassing both secondary and tertiary structure (Fig. 5.5-repaired). We conclude that the corrected alignment was more informative than the initial alignment for contact prediction.

We next measured the mean number of pairs in contact between the pair of positions with the n^{th} highest or better MI, Zp, ΔZp or Zpx values using the set of alignments in the CDD that met the minimum criteria, outlined in the Materials and Methods; these criteria are established as requirements to accurately identify contacting pairs in protein families. From these families, those that possessed 5 or more 6-residue segments with mean local Zp values ≥ 2.5 were selected. There were 16 such families that met the criteria for making contact predictions. These alignments are referred to as the 'worst CDD' because they are likely to contain systematic misalignments. The 84 protein families which were not included in the worst alignments set, which also met criteria for presence of covariation (Materials and Methods) are referred to as the 'best CDD' dataset. We also examined 5 highly curated protein families that were aligned using the Cn3D program [11] with multiple structural lines of evidence to support the inclusion of each sequence in the alignment (Materials and Methods).

The circles in Figure 5.6 show that all four covariation measures were able to identify many contacting pairs of positions in the curated alignments. MI was the worst performing measure, but the top 3 pairs were in contact in 4 of the 5 curated alignments. Zp was much better than MI, and ΔZp and Zpx outperformed Zp. The ΔZp and Zpx measures had similar accuracies; the top 7 highest scoring pairs of both ΔZp and Zpx were in contact and the top 20 pairs of

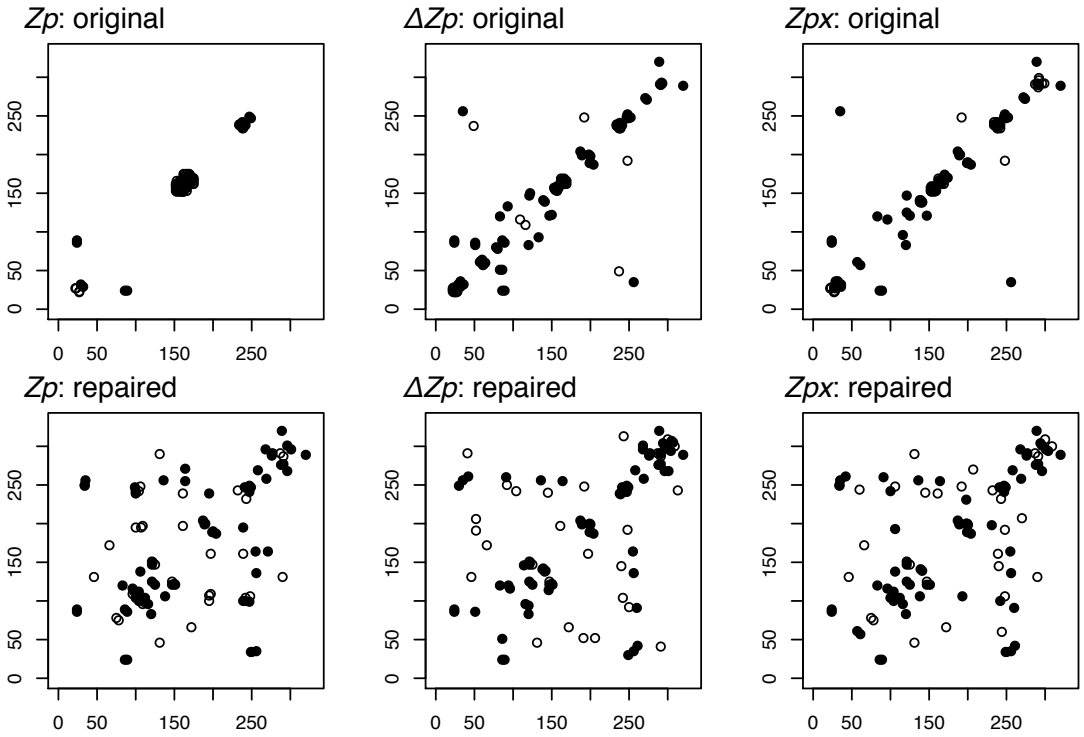

Figure 5.5: Predicted contact map shows repaired cd00300 alignment is more informative than the original. The top 50 highest Zp, ΔZp, and Zpx values were plotted on the 2-D map of the original cd00300 alignment (top) which contains systematic misalignments and the repaired version of cd00300 (bottom) with many misalignments removed. The data is displayed as a predicted contact map where black-filled circles are pairs in contact and white-filled circles are pairs not in contact. The majority of high-scoring Zp pairs in the original alignment are uninformative local contacts located in a major region of misalignment (top left). The contact maps of ΔZp and Zpx cover much more of secondary and tertiary structure in both the original and repaired alignments.

each had a ≥80% likelihood of contact.

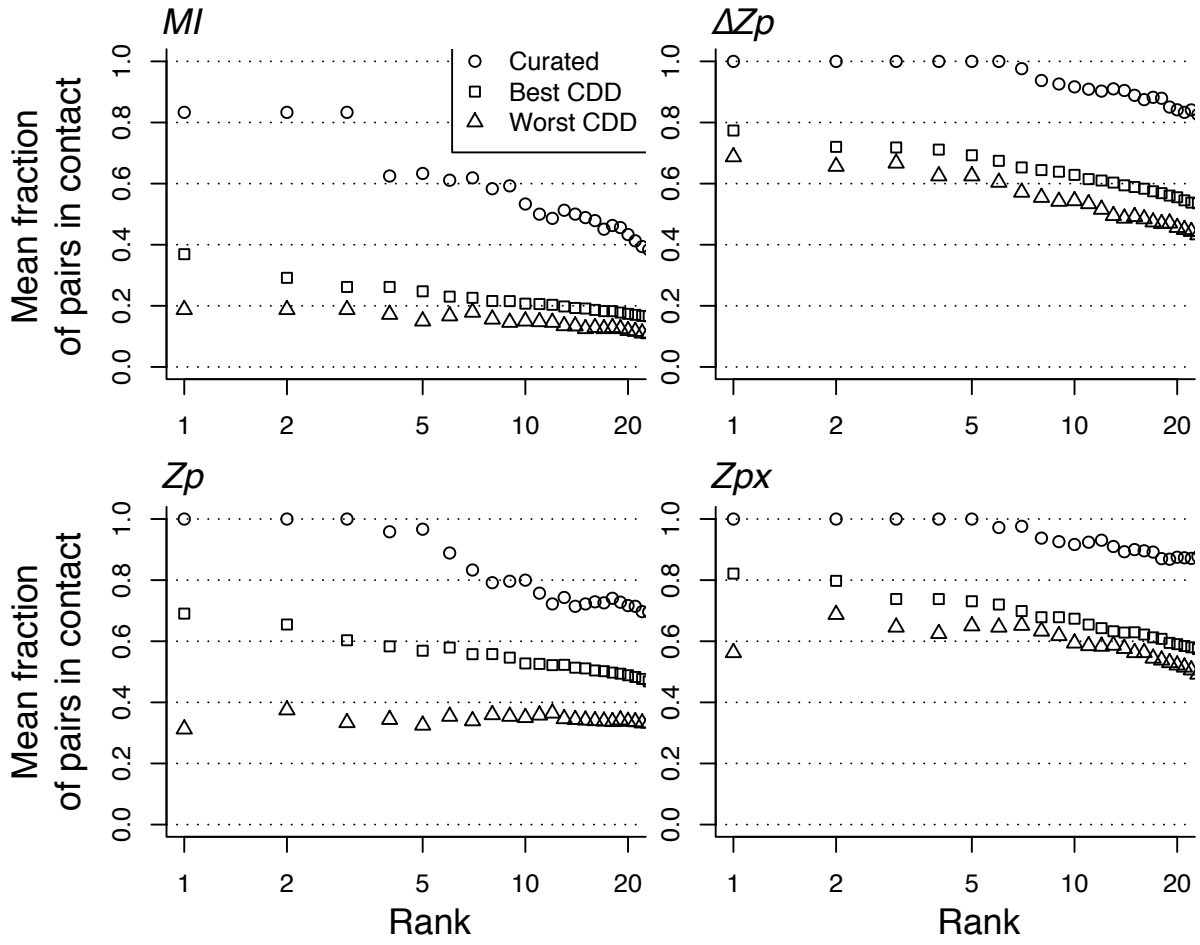

Figure 5.6: The effect of alignment quality on contact identification. All methods identify many contacts in the curated dataset. Plotted here is the mean fraction of pairs in contact of the top n^{th} ranked pairs (up to 20) for the Curated, Best CDD, and Worst CDD datasets using *MI*, *Zp*, ΔZp, and *Zpx*. The Curated dataset contains ideal alignments that are hand-curated for accuracy. Best CDD contains alignments which are unlikely to contain systematic misalignments. Worst CDD contains alignments which are likely to contain systematic misalignments. Only pairs 10 or more positions apart in sequence are included to prevent proximity in sequence from biasing results.

All four methods performed much worse in the CDD-based datasets. Again ΔZp and *Zpx* performed similarly and both were superior to *Zp*. However, *Zp* and *MI* identified very few positions in contact in the 'worst' alignments, and the highest scoring pair was no more likely to be in contact than the n^{th}-best scoring pair. We suggest that the large disparity between contact prediction accuracy of *Zp* on the best and worst datasets is because *Zp* is sensitivite to

systematic misalignments present in the 'worst' dataset while ΔZp and *Zpx* are not.

5.3.3 Comparison of sensitivity

We were interested in the sensitivites of *Zp*, ΔZp, and *Zpx*. The sensitivity of covariation methods is affected by the number of sequences [19], the number of positions in the alignment [3] and the structure used as the reference. The sensitivity of each method was determined by assessing the number of contacting pairs found at given likelihoods of contact, and is shown in Table S3 for contact likelihoods between 50% and 90%. All three methods identified over 17 pairs of positions per protein family with a contact likelihood of 50%. The number of pairs identified dropped off dramatically as the contact likelihood increased; *Zp* found an average of 4.4 pairs, ΔZp found an average of 5.6 pairs and *Zpx* identified 6.5 pairs at 90% contact likelihood. We conclude that *Zpx* is more sensitive than the other measures, but that no measure vastly outperforms any other.

We wanted to know if the three methods were identifying the same set or a different set of contacting pairs. We identified alignments in the CDD where $\geq$ 80% of the pairs found were in contact in each alignment (Materials and Methods, Table S4). *Zp* identified 767 pairs of which 85% were in contact, ΔZp identified 906 (85% contacting) and *Zpx* identified 1055 pairs (84% contacting). The overlap in the 1411 total pairs is shown in Figure 5.7, with the proportion of pairs in contact given below each measure. Several observations can be made. First, no method identified all pairs found by any other method; *Zpx*, ΔZp and *Zp* identified 75%, 64% and 54% of the total pairs with 15%, 14% and 10% of the pairs being unique to each method. The pairs most likely to be in contact were pairs identified by both ΔZp and *Zpx*. These 685 pairs composed 49% of the total pairs identified with 626 in contact (91% contacting). We conclude that pairs identified by both ΔZp and *Zpx* were more likely to be in contact than if either was used by itself.

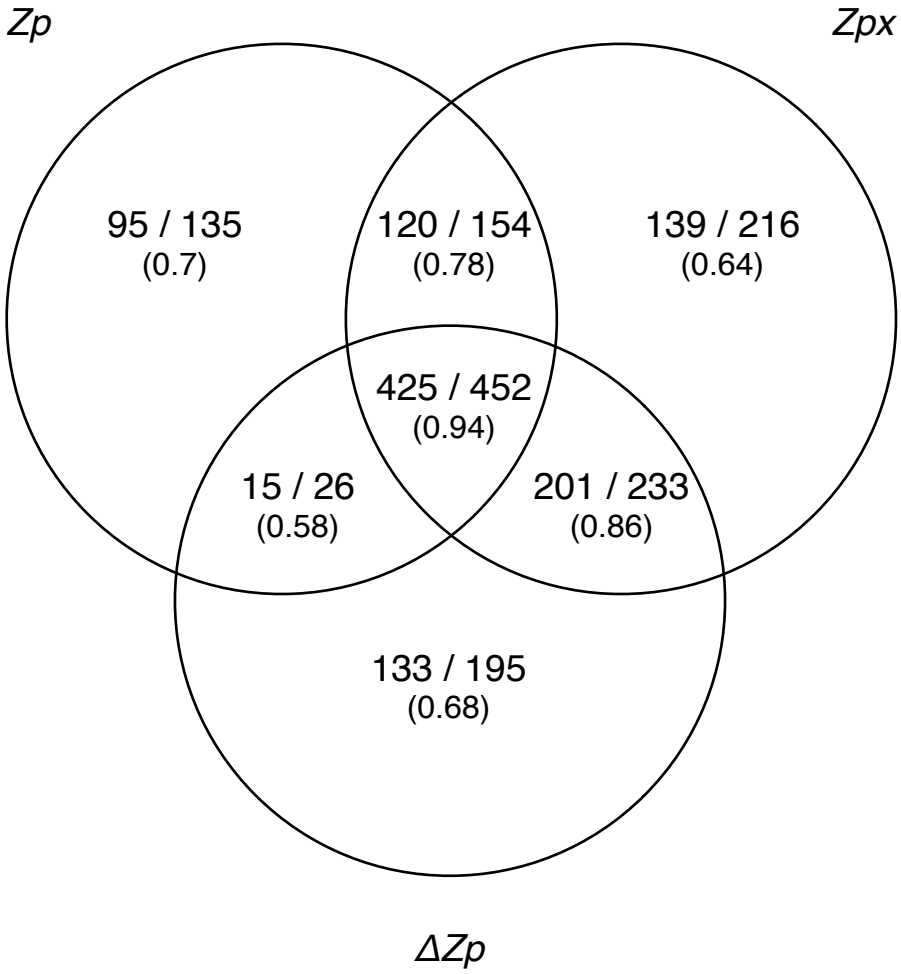

Figure 5.7: *Zp*, Δ*Zp*, and *Zpx* find different subsets of contacting positions. Venn diagram of pairs identified at 80% contact likelihood cutoff shown as number in contact vs. total predictions. Only pairs 10 residues or more apart in sequence were considered to prevent proximity in sequence from biasing results.

5.3.4 ΔZp and Zpx emphasize pairwise covariation

We were interested why ΔZp and Zpx were so effective at identifying contacting pairs. We examined this by modeling group coevolution with the simple model of *in silico* evolution described in the Materials and Methods and generated 10 independent alignments where the size of the coevolving group varied between 0 (no covariation) and 10, and the probability of coevolution was varied in increments between 0 and 0.95. These model alignments were used to examine the relationship between group size and the covariation score for each measure.

Figure 5.8 shows the effect of group size and coevolution probability on the mean values for the three statistics. If we examine the mean values for the extreme case where coevolution occurs at the maximum probability, we see that in all 3 methods the pairwise coevolving positions (ie. group size = 2) have much higher mean values than do instances where coevolution occurs in groups of 10. This effect is less pronounced for the intermediate group sizes of 4 or 5. In the case of Zp, residues that coevolve with an intermediate group size attain mean scores greater than 6, which is close to the mean Z score for coevolving pairs of positions. However, both ΔZp and Zpx show markedly lower mean scores for the intermediate sized groups of coevolving positions than for pairwise coevolving positions. The effect is similar at lower coevolution probabilities for all three methods, although it is non-linear for Zpx. We conclude that the ability of ΔZp and Zpx to emphasize the effect of pairwise covariation at the expense of group covariation explains in part why these two methods identify contacting pairs with greater sensitivity and specificity than other non-parametric methods.

5.4 Discussion

It is assumed that the covariation signal derived from structural and functional constraints is superimposed on the phylogenetic and stochastic signal [28]. This idea has led several groups to make the assumption that if a method identifies some structural covariation as indicated by contacting pairs, then other pairs with equivalent or higher scores *not in contact* must be

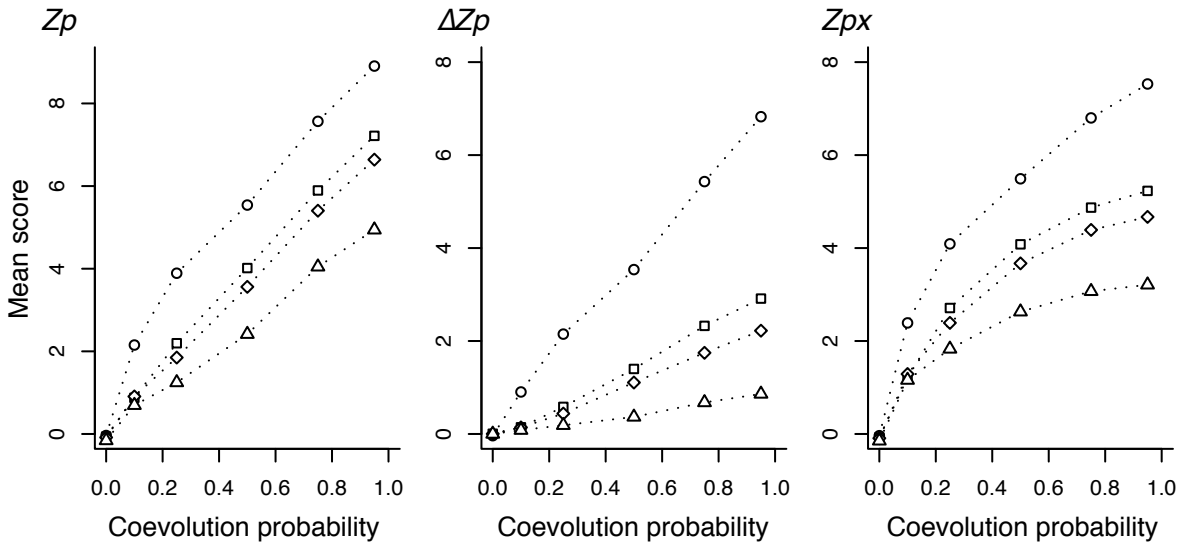

Figure 5.8: The effect of covariation probability and covarying group size on covariation measures. Sequences were evolved with each residue having a fixed probability of changing to another residue per time step as described in the Materials and Methods. Positions were placed in groups of 2 (◯), 4 (□), 5 (◇), or 10 (△) and constrained to coevolve with likelihoods of 0.1, 0.25, 0.5, 0.75 and 0.95. For example, if the group size was 4 and the likelihood of coevolution was 0.25, then if the residue of one member of the group was changed during the time step, then each of the other three members of the group was allowed to change to another residue with the fixed probability given on the X axis. The Y axis shows the mean score for each group size and each probability for 10 replicate *in silico* evolution runs.

caused by functional constraints [8, 16, 27]. However, since the true coevolving pairs in an alignment are unknown [28, 20], the alternative explanation is that some of these pairs may be false positive identifications. Here we show that a strong covariation signal can be caused by alignment error, potentially leading to false positive predictions. Specifically, covariation analysis of cd00300 would result in incorrect assignment of both conserved and covarying residues and thus an incorrect understanding of the protein.

Local covariation increases with misalignment proportionally to the amount of conservation at a position and inversely proportional to the amount of entropy. Covariation can be understood as proportionally high positional entropy relative to low joint entropy. This is related to the underlying information-theoretic values outlined in figure 5.3. Positions with low conservation are less susceptible to the local covariation effect because they already contain many of the possible residues making the proportional increase of entropy to joint-entropy less significant. Similarly, misaligning random sequences has a smaller effect than sequences that are misaligned together as a clade. If sequences are misaligned as a clade, the increase in joint-entropy is proportionally smaller because the sequences are related and aligned together which causes a larger increase in local covariation. The test to generate figures 5.1 and 5.2 is very conservative since the selection and shift are both done randomly. Conversely any algorithm which uses a phylogenetic tree to build the alignment will be susceptible to hierarchical clustering of misaligned sequences, which are easier to detect.

It is worth noting that the ability to detect misalignments is unique to Zp when compared to other statistics outlined in this manuscript. MI does not identify misalignments because background covariation signal is too high. The ΔZp and Zpx statistics do not identify mistalignments because they filter out the misalignment covariation signal. Zp works because it transforms the covariation values based on the assumption that the background covariation is due to shared phylogeny and relative entropy — an assumption that is explicitly violated when misalignments are introduced.

The increased local covariation methods outlined in this paper have already been critical

to the completion of two publications. In one [9], Gloor et al. used local covariation (as in Figure 5.2) to improve a structure-based sequence alignment of phosphoglycerate kinase. As outlined below, increased local covariation was the crucial tool that identified regions in the alignment which were likely to produce false positive results. In the second [13], Kleinstiver et al. used increased local covariation to validate an alignment of the GIY-YIG homing endonuclease I-Bmol, and prevented contamination by paralogous sequences. Furthermore, covariation statistics Zp, ΔZp, and Zpx were used to identify new structurally and functionally important pairs of residues. These successes demonstrate the effectiveness of increased local covariation and the new covariation statistics.

We found that flaws in the alignments themselves often result in positions having high covariation scores because of systematic misalignments. Since systematic misalignments involve several positions that are close in sequence (eg. cd00300), this could explain some of the group covarying positions that have been seen by many investigators, eg. [8, 5]. We suggest that evidence of group covariation between residues close in sequence be investigated carefully. For example, Gloor et al. found that increased local Zp identified two regions of phosphoglycerate kinase which contained subclusters of residues found in completely different environments [9]; structurally-conserved segments were either exposed to solvent or buried because of the replacement of a nearby alpha helix in some structures with a beta strand in others.

The logic of building an alignment is partially circular: alignments are built in part by maximizing sequence conservation, but then are used to find conserved positions which are, in turn, identified as important. While structure-based methods are often used as the benchmark and standard for protein alignments [25], Löytynoja and Goldman [17] showed that structure-based alignments are not as reliable as expected for genome annotation. Similarly, we found that some structure-based protein alignments are inappropriate for covariation analysis and, as noted above, that Zp can identify misaligned regions. We found that markedly elevated local Zp values were the hallmark of misaligned regions and suggest that the investigator proceeds with caution during the analysis of positions showing this pattern of covariation. If an investi-

gator draws conclusions from an alignment which has not been examined with increased local covariation, there is an increased risk of drawing erroneous conclusions from the alignment.

Finally, we demonstrated that Zpx and ΔZp are relatively insensitive to sequence misalignments explaining the increased sensitivity and selectivity of these methods to identify contacting pairs. The ability to identify misaligned segments coupled with the use of measures insensitive to misalignment reduces the risk of systematic misalignment and provides opportunities to correct and re-analyze the alignment. Zpx and ΔZp identified different subpopulations of pairs, which implies that neither method is, as yet, optimal. We find it interesting that both modifications use the independent covariation signal from positions i and j to derive the final statistic, suggesting that the relative covariation signal of each position is informative.

The sensitivity of Zp to systematic alignment errors can be exploited to identify regions of potential misalignment. Covariation provides an independent method for verifying the quality of an alignment and should therefore be especially useful for verifying alignments built on sequence conservation alone. The cd00300 alignment showed that misalignments occur in structure-based alignment datasets; importantly, cd00300 showed that misalignments occur in functionally-important conserved regions. We recommend investigating any incidents of increased local Zp (as in Figure 5.1, 5.2 or S1) as they may indicate systematic misalignment or an interesting phenomenon causing increased local covariation. We conclude that increased local covariation is an effective guide for improving or validating a multiple sequence alignment and initial observations suggest that mean pairwise Zp scores above 2.5 over a window of 6 should be investigated.

Our work shows that the quality of the alignment is critical for correct assignment of pairs of residues in covariation analyses; unfortunately, all alignment methods produce lower-quality covariation predictions when erroneous sequences are included. However, it is impossible to know for certain if all positions in an alignment are assigned correctly even when using state-of-the-art methods. Therefore, we recommend ΔZp and Zpx over other statistics as they provide better contact prediction because they are demonstrably less susceptible to systematic

misalignment errors than other covariation measures like *Zp*.

5.5 Materials and Methods

5.5.1 Modeling Systematic Misalignment

A hand-curated alignment for triosephosphate isomerase was created using Cn3D [11]. An ungapped segment of the alignment was selected to be the artificially misaligned segment. The misaligned segment is highlighted in each figure. Within this segment, each sequence has a 5%, 10%, or 30% chance of being shifted. Each sequence selected to be misaligned has an equal chance of being shifted one position left or right.

5.5.2 Alignment curation and criteria for contact prediction

Multiple sequence alignments were extracted from the CDD database downloaded from NCBI on April 17, 2008. They were curated to include only those alignments with at least one structure, more than 125 sequences and ≥50 ungapped positions in the alignment. Three datasets were benchmarked in Figure 5.6. 'Worst CDD' are a subset of the outlined CDD sequences which are likely to contain misalignments as they contain 5 or more 6-residue segments with mean $Zp \geq 2.5$. 'Best CDD' contains sequences which are not in the 'Worst CDD' dataset, but which also have at least (L / 10) values of $Zp \geq 4.5$ (where L is the length of the protein). This ensures that the alignment has some covariance information, but makes no judgements about the location of these pairs in sequence or in structure. The 'Curated' dataset contains 5 structure-based hand-curated alignments curated according to Dunn et al. [3]. For Figure 5.7, we identified alignments in the CDD that had ≥ 150 sequences, ≥ 50 non-gapped positions. We curated this set so that each covariance method was able to predict contacting pairs at an accuracy of at least 80%. To ensure the structure used to assign contacting pairs did not bias the results, we used alignments where the covariance methods agreed on which structure was

of highest quality. The 100 alignments meeting these criteria are listed in Table S4. When covariation statistics predict contacting pairs, we define contact as any non-hydrogen atom from one residue being within 6 Å of any non-hydrogen atom from the other residue. To prevent proximity in sequence from biasing results, only positions 10 or more apart in sequence are considered when using contact as a benchmark.

5.5.3 cd00300-based alignments

cd00300 is a structure-based alignment of lactate dehydrogenase from CDD. The original cd00300 dataset is composed of the sequences in this alignment. The repaired cd00300 dataset has 13 sequences removed because of alignment issues (annotated in Table S2). The original cd00300 alignment is used as it existed in CDD. The repaired version of cd00300 was realigned using Cn3D [11] based on a refined structure alignment based on errors of probable misalignment according to increased local covariance.

5.5.4 Covariance statistic calculations

Covariance statistics were calculated according to Martin et al. [19], and Dunn et al. [3]. *Zpx* is similar to the *Zres* statistic of Little and Chen [15], but simpler to calculate (Supplementary Methods S1). Positions that contain gaps are not analyzed because gaps violate the assumption of orthology connecting covariation with coadaptation and coevolution (Figure S2).

Mutual Information (MI) measures the reduction in uncertainty of one variable given information about another variable and was calculated as previously [19]. As shown in equation 1, in the context of protein sequence families it measures the difference between the expected entropy (H) of residues in two columns i and j if they were independent against the observed joint entropy, $H_{i,j}$.

$$MI_{i,j} = H_i + H_j - H_{i,j} \tag{5.1}$$

The underlying assumption when calculating MI is that all events are independent; this is not

true in the context of protein families since, to a first approximation, every position in a protein family shares common ancestry with every other position, and the positions in a gene for a given protein are rarely split by recombination. *MIp*, the product corrected *MI*, estimates the background *MI* signal caused by sequence non-independence [3] as shown in equation 2

$$MIp_{i,j} = MI_{i,j} - (\overline{MI}_{i,x} \times \overline{MI}_{j,x})/\overline{MI} \tag{5.2}$$

where $\overline{MI}_{i,x}$ is the mean *MI* of position i with all other positions and $\overline{MI}$ is the overall mean *MI*.

The *MIp* values were converted to Z scores since absolute *MIp* values vary somewhat between alignments and because the underlying distribution of *MIp* values approximates a Gaussian distribution in the absence of structural and functional covariation [3]:

$$Zp_{i,j} = (MIp_{i,j} - \overline{MIp})/\sigma(MIp) \tag{5.3}$$

where again $\overline{MIp}$ is the mean *MIp* and $\sigma(MIp)$ is its standard deviation.

Zpx is a modification of Little and Chen's *Zres* statistic [15] that is based on the residual *MI* of a linear regression between $MI_{i,j}$ and the mean *MI* of positions i and j, $\overline{MI_i} \times \overline{MI_j}$. The plot shown in Figure 1A shows that the residual is nearly identical to *MIp* with a slope of 1, an intercept of 0 and an r^2 value of 0.9995. *Zres* is the product of the Z scores derived from the residuals for each individual position [15]. Since the residual and *MIp* are virtually indistinguishable (Supplementary Methods S1), we substituted the more efficiently calculated *MIp* for the residual in the formula for *Zres* as shown in equation 5 to calculate $Z_{i\times j}$.

$$Z_{i\times j} = \frac{MIp_{i,j} - \overline{MIp_i}}{\sigma(MIp_i)} \times \frac{MIp_{ij} - \overline{MIp_j}}{\sigma(MIp_j)} \tag{5.4}$$

As expected from the similarity of the underlying statistics, *Zres* and $Z_{i\times j}$ are extremely similar with a slope of 0.9998, an intercept of 0.0005 and an r^2 value of 0.9998. Note that the

$Zres$ and $Z_{i\times j}$ values are the product of the two Z scores, and so these values scale geometrically when compared to Zp. In this study we use Zpx, which is the square root of $Z_{i\times j}$ to allow comparison of the Zp and $Z_{i\times j}$ values on similar scales.

ΔZp is a measure of the difference score between successive Zp scores at position i. ΔZp was calculated by placing the Zp values of position i with all other positions, x, in an ordered list, with the largest Zp value first, i.e. $L_i = (Zp_{i,x}, Zp_{i,x+1}, Zp_{i,x+2}...Zp_{i,x+(N-1)})$, where N is the number of Zp values for position i. ΔZp is the sequential difference between list elements scaled by interquartile units as follows:

$$\Delta Zp_{i,x} = \frac{Zp_{i,x} - Zp_{i,x+1}}{R} \tag{5.5}$$

where R is the interquartile range calculated as $\frac{1}{2}$ the difference between the 75^{th} and the 25^{th} percentile for the data in L_i. The interquartile range scaling factor and the median provide robust estimates of dispersion and central tendency that compensate for variation in the distribution of Zp at each position while making the minimum number of assumptions about the underlying distribution. Thus ΔZp measures how extreme the difference in Zp scores is for all pairs of positions, i, x.

Since each Zp score is a measure of the covariation between two positions, i and j, there are two ΔZp scores for each pair of positions: one is the difference between $Zp_{i,j}$ and the next highest score for position i, and the other is the difference between $Zp_{j,i}$ and the next highest scoring position for position j. We used the greater of the ΔZp scores.

MI, and the derived statistics, *MIp*, ΔZp and Zpx were calculated only for *ungapped* positions in the multiple sequence alignments. Covariation analysis attempts identify those positions that are coevolving for structural or functional reasons, and as shown in this report, depends upon the precise placement of homologous positions in the alignment.

5.5.5 Screening for misalignments using increased local MIp

The average Zp score and average entropy were calculated for all pairs in a ungapped window of width 6. When graphed, high peaks represent increased local Zp. We consider peaks of height 2.5 or higher to be worth investigating, but these could also represent strong covariation due to secondary structure.

5.5.6 Synthetic coevolution dataset

The data in Figure 5.8 were generated using the simple model of coevolution described previously [19]. In brief, the initial sequence had substitutions introduced at each position with a probability derived from a uniform probability distribution. Sequences were split or 'speciated' with a constant probability. Groups of positions were constrained to coevolve such that if a substitution occurred in one member of the group the remaining members of the group had a substitution introduced with a given probability which ranged between 0.1 to 0.95. Group sizes were varied between 2 and 10. Alignments derived from this method have been shown to recapitulate many properties relevant to coevolution [19].

Bibliography

[1] W R Atchley, K R Wollenberg, W M Fitch, W Terhalle, and A W Dress. Correlations among amino acid sites in bhlh protein domains: an information theoretic analysis. *Mol Biol Evol*, 17(1):164–178, 2000.

[2] Cristina Marino Buslje, Javier Santos, Jose Maria Delfino, and Morten Nielsen. Correction for phylogeny, small number of observations and data redundancy improves the identification of coevolving amino acid pairs using mutual information. *Bioinformatics*, 25(9):1125–1131, 2009.

[3] SD Dunn, LM Wahl, and GB Gloor. Mutual information without the influence of phylogeny or entropy dramatically improves residue contact prediction. *Bioinformatics*, 23(3):333–340, 2008.

[4] Robert C Edgar and Serafim Batzoglou. Multiple sequence alignment. *Curr Opin Struct Biol*, 16(3):368–73, Jun 2006.

[5] Mario A Fares and Simon A A Travers. A novel method for detecting intramolecular coevolution: adding a further dimension to selective constraints analyses. *Genetics*, 173(1):9–23, 2006.

[6] W M Fitch and E Markowitz. An improved method for determining codon variability in a gene and its application to the rate of fixation of mutations in evolution. *Biochem Genet*, 4(5):579–593, 1970.

[7] CA Floudas, HK Fung, SR McAllister, M Monnigmann, and R Rajgaria. Advances in protein structure prediction and de novo protein design: A review. *Chemical Engineering Science*, 61(3):966–988, February 2006.

[8] Gregory B Gloor, Louise C Martin, Lindi M Wahl, and Stanley D Dunn. Mutual information in protein multiple sequence alignments reveals two classes of coevolving positions. *Biochemistry*, 44(19):7156–7165, 2005.

[9] Gregory B Gloor, Gaurav Tyagi, Dana M Abrassart, Andrew J Kingston, Andrew D Fernandes, Stanley D Dunn, and Christopher J Brandl. Functionally compensating, coevolving positions are neither homoplasic nor conserved in clades. *Mol Biol Evol*, Jan 2010.

[10] O Gotoh. Significant improvement in accuracy of multiple protein sequence alignments by iterative refinement as assessed by reference to structural alignments. *J Mol Biol*, 264(4):823–38, Dec 1996.

[11] C W Hogue. Cn3d: a new generation of three-dimensional molecular structure viewer. *Trends Biochem Sci*, 22(8):314–6, Aug 1997.

[12] Changhoon Kim and Byungkook Lee. Accuracy of structure-based sequence alignment of automatic methods. *BMC Bioinformatics*, 8:355, 2007.

[13] Benjamin P Kleinstiver, Andrew D Fernandes, Gregory B Gloor, and David R Edgell. A unified genetic, computational and experimental framework identifies functionally relevant residues of the homing endonuclease i-bmoi. *Nucleic Acids Res*, Jan 2010.

[14] Rachel Kolodny, Patrice Koehl, and Michael Levitt. Comprehensive evaluation of protein structure alignment methods: scoring by geometric measures. *J Mol Biol*, 346(4):1173–88, Mar 2005.

[15] Daniel Y Little and Lu Chen. Identification of coevolving residues and coevolution potentials emphasizing structure, bond formation and catalytic coordination in protein evolution. *PLoS ONE*, 4(3):e4762, 2009.

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

[17] Ari Löytynoja and Nick Goldman. Phylogeny-aware gap placement prevents errors in sequence alignment and evolutionary analysis. *Science*, 320(5883):1632–5, Jun 2008.

[18] Aron Marchler-Bauer, John B Anderson, Farideh Chitsaz, Myra K Derbyshire, Carol DeWeese-Scott, Jessica H Fong, Lewis Y Geer, Renata C Geer, Noreen R Gonzales, Marc Gwadz, Siqian He, David I Hurwitz, John D Jackson, Zhaoxi Ke, Christopher J Lanczycki, Cynthia A Liebert, Chunlei Liu, Fu Lu, Shennan Lu, Gabriele H Marchler, Mikhail Mullokandov, James S Song, Asba Tasneem, Narmada Thanki, Roxanne A Yamashita, Dachuan Zhang, Naigong Zhang, and Stephen H Bryant. Cdd: specific functional annotation with the conserved domain database. *Nucleic Acids Res*, 37(Database issue):D205–10, 2009.

[19] L C Martin, G B Gloor, S D Dunn, and L M Wahl. Using information theory to search for co-evolving residues in proteins. *Bioinformatics*, 21(22):4116–4124, 2005.

[20] R Oliveira and A Pedersen. Finding coevolving amino acid residues using row and column weighing of mutual information and multi .... *Algorithms for molecular biology*, Jan 2007.

[21] Florencio Pazos and Alfonso Valencia. Protein co-evolution, co-adaptation and interactions. *EMBO J*, 27(20):2648–2655, 2008.

[22] D D Pollock, W R Taylor, and N Goldman. Coevolving protein residues: maximum likelihood identification and relationship to structure. *J Mol Biol*, 287(1):187–198, 1999.

[23] G Shackelford and K Karplus. Contact prediction using mutual information and neural nets. *Proteins*, 2007.

[24] Michael Socolich, Steve W Lockless, William P Russ, Heather Lee, Kevin H Gardner, and Rama Ranganathan. Evolutionary information for specifying a protein fold. *Nature*, 437(7058):512–518, 2005.

[25] Julie D Thompson, Patrice Koehl, Raymond Ripp, and Olivier Poch. Balibase 3.0: latest developments of the multiple sequence alignment benchmark. *Proteins*, 61(1):127–36, Oct 2005.

[26] Elisabeth R M Tillier and Thomas W H Lui. Using multiple interdependency to separate functional from phylogenetic correlations in protein alignments. *Bioinformatics*, 19(6):750–755, 2003.

[27] Simon A A Travers and Mario A Fares. Functional coevolutionary networks of the hsp70-hop-hsp90 system revealed through computational analyses. *Mol Biol Evol*, 24(4):1032–1044, 2007.

[28] K R Wollenberg and W R Atchley. Separation of phylogenetic and functional associations in biological sequences by using the parametric bootstrap. *Proc Natl Acad Sci U S A*, 97(7):3288–3291, 2000.

[29] Karen M Wong, Marc A Suchard, and John P Huelsenbeck. Alignment uncertainty and genomic analysis. *Science*, 319(5862):473–6, Jan 2008.

[30] C Yanofsky, V Horn, and D Thorpe. Protein structure relationships revealed by mutational analysis. *Science*, 146:1593–1594, 1964.

[31] Kevin Y Yip, Prianka Patel, Philip M Kim, Donald M Engelman, Drew McDermott, and Mark Gerstein. An integrated system for studying residue coevolution in proteins. *Bioinformatics*, 24(2):290–292, 2008.

Chapter 6

Coevolution statistics detect benchmark alignment errors

6.1 Abstract

The use of sequence alignments to understand protein families is ubiquitous in molecular biology. High quality alignments are difficult to build and protein alignment remains one of the largest open problems in computational biology. Misalignments can lead to inferential errors about protein structure, folding, function, phylogeny, and residue importance. Identifying alignment errors is difficult because alignments are built and validated on the same primary criteria: sequence conservation. Local covariation identifies systematic misalignments and is independent of conservation.

We demonstrate an alignment curation tool, LoCo, that integrates local covariation scores with the Jalview alignment editor. Using LoCo, we illustrate how local covariation is capable of identifying alignment errors due to the reduction of positional independence in the region

A version of this chapter has been published.
RJ Dickson, GB Gloor. (2012). Protein Sequence Alignment Analysis by Local Covariation. PLoS One. 7(6): e37645.

of misalignment. We highlight three alignments from the benchmark database, BAliBASE 3, that contain regions of high local covariation, and investigate the causes to illustrate these types of scenarios. Two alignments contain sequential and structural shifts that cause elevated local covariation. Realignment of these misaligned segments reduces local covariation; these alternative alignments are supported with structural evidence. We also show that local covariation identifies active site residues in a validated alignment of paralogous structures.

Loco is available at https://sourceforge.net/projects/locoprotein/files/

6.2 Introduction

Multiple sequence alignments are critical for generating and testing hypotheses based on protein structure, function, and phylogeny. Protein alignments are built based on the assumption that each position (column) in the alignment is homologous [11]. With structural information, homology is typically validated by demonstrating that two residues occupy the same location in 3D space since structural homology implies sequential homology [24]. If only sequence information is available, positions are assigned based on the conservation of residue identity or properties, which is inherently less reliable than structural inference. The logic of interpreting sequence alignments is, therefore, circular: alignments are built, validated, and used based on a single criterion, conservation. A conservation-independent property of sequence alignments is a valuable adjunct to validate a sequence alignment.

Structure alignments are used to validate sequence alignments because they provide evidence independent of sequence; thus, benchmark datasets like BAliBASE include structural support [37, 36]. Unfortunately, structures are comparatively rare and cannot be used to validate all sequence alignments. In BAliBASE 3, there are many alignments that contain few structural seeds compared to the number of sequences. Furthermore, Kuziemko et al. noted that structurally supported alignments often do not score as highly as alignments that optimize the dynamic programming scoring function of sequence alignment algorithms, suggesting se-

quence alignment algorithms frequently reject the structurally valid alignment when such an alignment exists [24]. As sequence and structure grow more distant it becomes increasingly difficult to produce an alignment.

Multiple sequence alignment methods are typically benchmarked against high-quality datasets such as BAliBASE [37, 36]. In principle, BAliBASE alignments should represent the upper limit of quality that can be achieved using existing methods as they are both structure-aided and manually curated. Authors of sequence alignment algorithms strive to create alignments that are the most similar to the benchmark dataset. Benchmark datasets must be of the utmost quality to be reliable for assessing competing methods. However, Edgar demonstrated that inconsistencies and potential errors exist even in benchmark datasets like BAliBASE [9].

Another resource for hand-curated structure-based sequence alignments is the Conserved Domain Database (CDD) [28]. While CDD, was not originally designed to be a benchmark dataset like BAliBASE 3, its hand-curated structure alignments of are sufficient quality to be used as the benchmark dataset when analyzing structure alignment algorithms [22].

Alignments are also susceptible to errors for reasons independent of the circular logic of sequence alignment. Without careful manual curation, structure alignment algorithms are susceptible to shift error [22]. Shift errors are misalignments where the sequence has been shifted by 1 or more positions, though major secondary structural elements are still aligned. Since structure-based alignments are built by progressively aligning sequences to the seed structure alignment, shift errors can propagate systematically.

Progressive multiple sequence alignment strategies can also be prone to systematic propagation of errors as sequences are progressively added to a growing alignment. Iterative sequence alignment methods attempt to resolve this issue by employing a refinement step after the initial alignment is built. However, at present there is no method that reliably identifies shift errors. A disagreement between two theoretically valid alignment predictions is therefore very difficult to resolve; the current solution is to trust a benchmark dataset if available.

Covariation analysis is a statistical method used to understand coevolution in proteins [1].

Covariation can be understood intuitively as a measure of the reduction in uncertainty about one position given information about another. Covariation scores have minima when either both positions are absolutely conserved or when both positions are randomly assorting. A high covariation score implies that knowledge of one position provides information about the identity of the other.

Covariation statistics are used to indicate whether two residues are potentially coevolving [12, 30, 20, 10, 7, 26, 6, 33]. Coevolving residues are thought to arise by a mechanism of constrained amino acid change [12, 40, 32]. Many covariation statistics predict contacting pairs with high accuracy [7, 6, 26, 39]. If this dependency between positions is due to some evolutionary process, like structural or functional constraints, then it is often defined as coevolution [2]. For clarity, coevolution is an evolutionary process, and covariation is the statistical non-independence used to identify it. When using covariation statistics to find coevolving pairs of positions a number of assumptions about the nature of the alignment are made; this includes the assumption that the protein family is properly aligned and all members are orthologous [6].

We previously demonstrated that with systematic sequence shifts (ie. synthetic misalignments), alignments show patterns of increased sequence-local covariation in the shifted segment [6]. We have extended the observations from [6] into an alignment curation tool called LoCo. LoCo is based on Jalview [4, 38] and provides a local covariation measure in real-time while curating an alignment. We use case studies to show how to apply LoCo to both the Conserved Domain Database [22] and BAliBASE 3 database [36] to identify sequence alignments that have regions of high local covariation. We provide examples of structurally validated realignments of the BAliBASE 3 benchmark dataset with both covariation and structural justification. Increased local covariation also identifies important functional residues in a structurally valid alignment from the BAliBASE 3 database. Finally, we demonstrate the method of investigating local covariation to determine if adjustments of the alignment is warranted.

6.3 Results

6.3.1 Illustrating How Covariation Identifies Sequence Shifts

The covariation statistic *Zp* (calculated as in Materials and Methods) is exquisitely sensitive to identifying residue non-independence in pairs of columns [7, 6]. To illustrate this effect, we created a 7-position synthetic alignment prepended to a 200 residue alignment of methionine aminopeptidase (Materials and Methods). Each column in the alignment is composed of a random assortment of 3 residues. Then, a small fraction of positions two through six were shifted 1 position to the right (Figure 6.1A). Positions 1 and 7 were not shifted and so were always randomly assorting relative to the other positions.

This alignment was loaded into the LoCo alignment viewer, which uses the existing Jalview

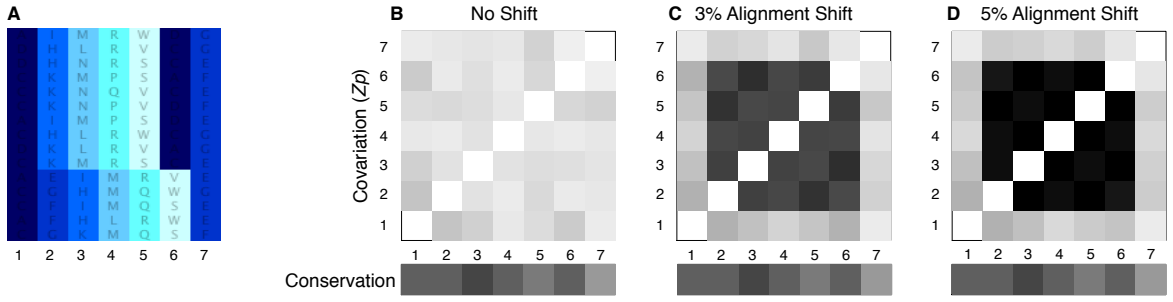

Figure 6.1: Local covariation identifies alignment shift errors. (**A**) A synthetic alignment was created for covariation analysis. Each of the 7 positions (columns) in the alignment contained a random assortment of 3 residues. A subset of the sequences (rows) in the alignment were then shifted for positions 2, 3, 4, 5, and 6 one position to the right. Position 1 and position 7 were not shifted. (**B**) A matrix of all pairwise covariation scores for the unshifted synthetic alignment where darker grey represents higher covariation calculated as in Materials and Methods. All positions randomly assort compared to one another; thus, panel **B** represents the background covariation for the synthetic block. Jalview conservation scores are also shown for each position. (**C**) Matrix of covariation scores where 3% of sequences (6 of 200) are shifted for positions 2 — 6. Covariation increases between all shifted positions, but does not increase between unshifted positions 1 and 7 and any of the unshifted positions. Conservation scores remain unchanged. (**D**) Matrix of covariation scores like panel **C**, except 5% of sequences are shifted. Covariation scores increase further between shifted positions, but unshifted positions show scores comparable to background as in panel **A**. Conservation scores remain unchanged.

codebase but replaces the Quality score with Local Covariation (Materials and Methods). Figure 6.1B (top) shows a heatmap of covariation scores when the statistic *Zp* [7] is applied to the synthetic block when no sequences are shifted (Materials and Methods). Darker shading represents higher conservation or covariation scores. Since all positions are randomly assorting and thus independent of one another, this heatmap represents the background covariation. The starting conservation scores, as calculated by Jalview, for the initial aligned positions are shown below. Figure 6.1B establishes a baseline for comparison; light grey implies a negligible covariation score.

The heatmap shown in Figure 6.1C shows all pairwise covariation scores when positions 2 through 6 contain 3% (6 of 200) shifted sequences. It is apparent that all pairwise covariation scores in the shifted region have increased compared to the baseline. Furthermore, the unshifted flanking positions, 1 and 7 (and all other unshifted positions in the MAP1 alignment), remain unchanged compared to the baseline shown in Figure 6.1B and have negligible covariation scores. Finally, Figure 6.1D shows that when 5% (10 of 200) of sequences are shifted, there is a marked increase in covariation scores in the misaligned region; also, there is no noticeable change in the amount of covariation between any unshifted positions. Finally, notice that conservation, which is the primary criterion on which alignments are built and evaluated, remains visibly unchanged in Figures 6.1B, 6.1C, and 6.1D.

The reason for increased local covariation in the shifted regions is the reduction of uncertainty between shifted positions[7, 6]. When two positions assort independently, as seen in Figure 6.1B, the knowledge of the residue present at a given position provides no information about any other position. However, when a block of sequence is shifted, positions are no longer independent, and positions in the same shifted block share predictive power. This illustration explains the observation in [6] that local covariation strongly correlates with systematic misalignments.

This simple illustration shows that local covariation easily identifies segments of alignments with these types of sequence shifts as described previously [6]. Previously [14], we used

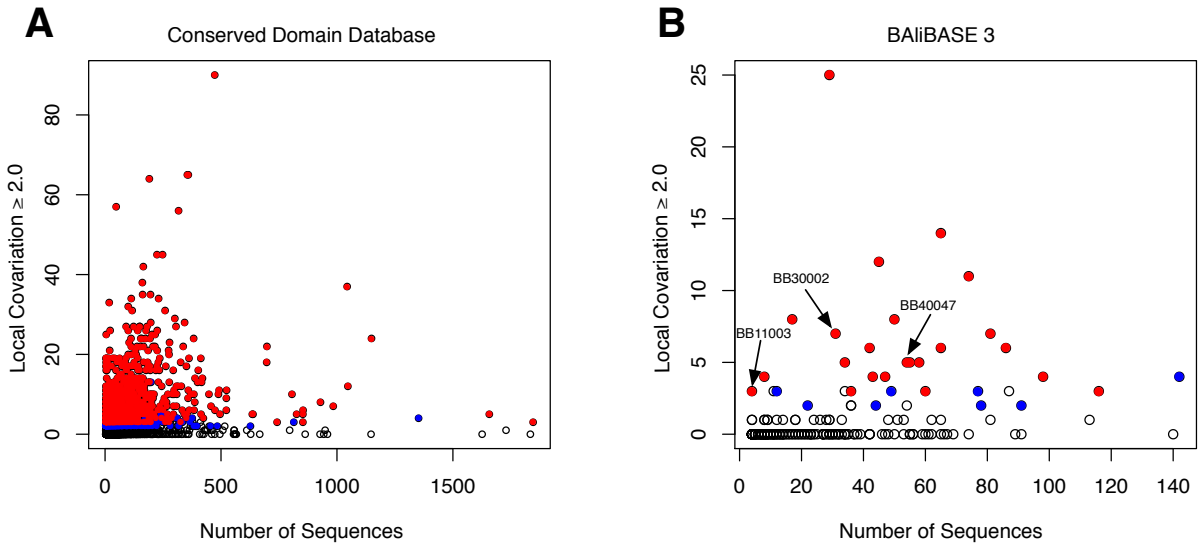

Figure 6.2: Alignments with high local covariation found in alignment databases. Each alignment in the Conserved Domain Database [28] and BAliBASE 3 [36] is represented by a single circle. Alignments are partitioned by the number of sequences and the number of regions of high local covariation. A region of high local covariation is defined as a local covariation peak greater than or equal to 2.0. Alignments with two adjacent regions of high local covariation are coloured blue. Regions that contain three or more contiguous regions of high local covariation are coloured red. (**A**) Analysis of all conserved domains (cd) in the Conserved Domain Database (CDD). (**B**) Analysis of all alignments in BAliBASE 3.

local covariation to identify a region that could assume either be alpha helical or beta stranded conformation within the orthologous phosphoglycerate kinase gene family. The remainder of this paper shows how local covariation can be used to identify other possible sources of high local covariation. As shown here, these can include putative systematic sequence misalignments and paralogous contamination of gene families.

6.3.2 Identifying Alignments with High Local Covariation

Local covariation is calculated as the mean covariation score over a window 6. If this mean score is greater than or equal to 2.0 then it is considered a high local covariation peak. Of the 6874 conserved domains (cd) analyzed in the CDD database, 2189 had at least one peak at or above 2.0 (Figure 6.2A). We also analyzed the BAliBASE 3 benchmark database. Figure

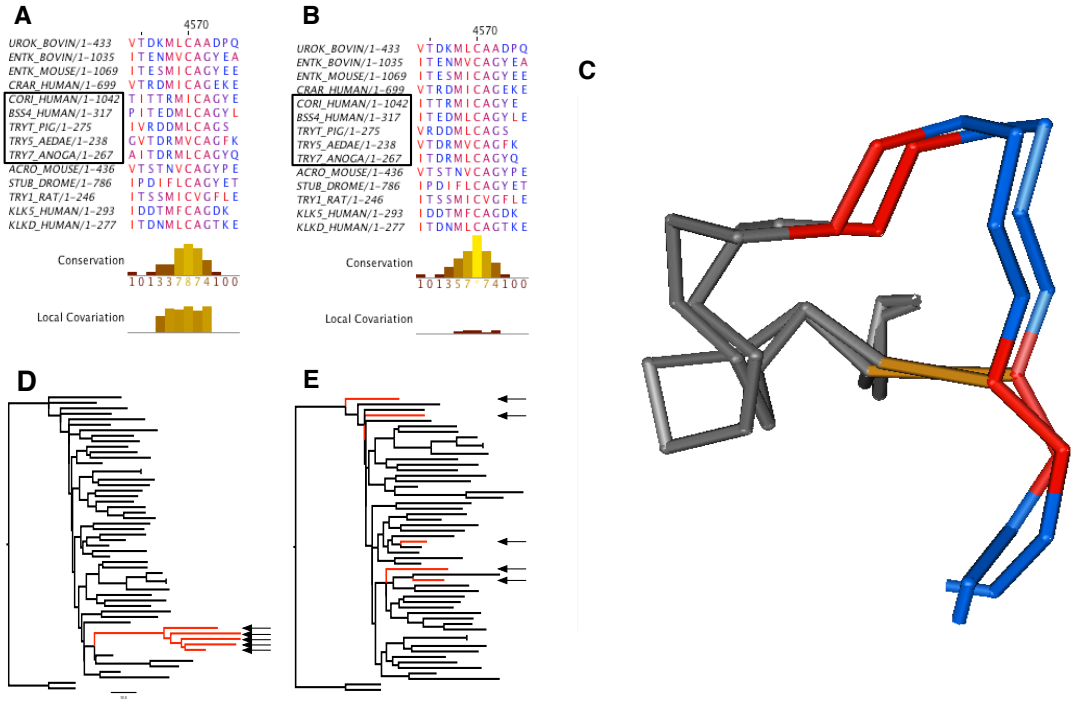

Figure 6.3: Realigning serine protease using LoCo. (**A**) Region of high local covariation and good conservation from alignment BB40047 from the BAliBASE 3 benchmarking dataset [36]. Five highlighted sequences do not show conservation of the disulphide bonded cysteine shown structurally in panel **C**. (**B**) Realignment of region from panel **A** using local covariation as a guide. (**C**) Structural validation of the alignment from panel **B** built in Cn3D [18]. Positions homologous to those shown in panels **A** and **B** are coloured by identity; the conserved disulphide bond is highlighted in orange. (**D**) Neighbour joining tree of high local covariation segment shown in panel **A**. Potentially misaligned sequences (indicated by arrows) cluster in a clade joined to the remainder by a long branch. (**E**) Neighbour joining tree based on realigned segment in **B** shows realigned sequences no longer cluster together as expected by the phylogenetic relationship of the organisms.

6.2B shows that the majority of BAliBASE alignments do not have regions of increased local covariation. However, we found that 60 of the 217 alignments in BAliBASE 3 had at least one peak at or above the 2.0 local covariation threshold. Regions of high local covariation appear to be common in these alignment databases. We show below that these should be investigated manually to determine the root cause.

6.3.3 Realigning a BAliBASE Multiple Sequence Alignment

In the BAliBASE 3 dataset, there were 37 alignments that contain contiguous blocks of high local covariation (filled dots). Of these, 30 had three or more contiguous high local covariation peaks, representing an extended range of high local covariation. We have chosen 3 alignments BB11003, BB30002, and BB40047 from 3 different categories of BAliBASE that demonstrate the characteristics of alignments with high local covariation. We illustrate how LoCo can be used to characterize the source of the high local covariation.

We identified a contiguous segment of high local covariation in the BB40047 alignment of BAliBASE 3. BB40047 is built upon the alignment of two structures containing a disulphide bond shown in Figure 6.3C. Figure 6.3A, a screenshot from the LoCo tool, shows the sequence alignment corresponding to the coloured region of the structure in Figure 6.3C. The region highlighted is the only block showing increased local covariation in this alignment. The BAliBASE alignment does not show conservation of the disulphide bonded cysteine; the presence of a cysteine is necessary to maintain the disulphide bond.

Although there is no structural information for the highlighted sequences, we can infer that the adjacent cysteine should be aligned to the disulphide bonded position because the existing alignment would place the cysteine in a conformation unable to form a disulphide bond. Figure 6.3D shows that the highlighted sequences group together when the region of high local covariation is clustered using the built-in Jalview function for neighbour joining tree by percent identity. Using the procedure outlined in the Methods section, the region can be adjusted as shown in Figure 6.3B. The adjusted alignment shows perfect conservation of the cysteine that is absolutely necessary for maintaining the disulphide bond shown structurally in Figure 6.3C. The adjusted alignment also shows a marked decrease in local covariation. After the sequences have been adjusted, they no longer cluster together (Figure 6.3E). Instead, clustering is more similar to that expected by the organism relationships.

Thangudu et al. noted that imperfect conservation of disulphide bonds in alignments is frequently caused by structure or sequence alignment errors [35]. The decrease in local covari-

ation comparing the original BAliBASE (Figure 6.3A) with the realigned (Figure 6.3B) and the absolute conservation of the disulphide bond illustrates how LoCo can be used for identifying potentially troublesome sites.

6.3.4 Realigning a BAliBASE Structure Alignment

Some structure alignments generated by unsupervised algorithms suffer from shift error [22] where the location of secondary structures are aligned correctly, but pairwise alignment of residues is offset relative to the periodicity of the secondary structural element. Such alignments can be difficult to identify visually or by root-mean-square deviation (RMSD) because the unshifted alignment preceding and following the misalignment can create the appearance of correct alignment; as well, the misalignment can be obscured by other structures. We demonstrate this type of an erroneous structure alignment in Figure 6.4.

Local covariation analysis of BAliBASE 3 identified a region of interest in the alignment BB30002 (Figure 6.2). BB30002 is particularly difficult to analyze because it is an alignment of several paralogous tRNA synthetases. In intra-molecular coevolution analyses, paralogous sequences are seen as contamination and can lead to false-positve conclusions since their presence violates the implicit assumptions of coevolutionary analyses [23]. The BAliBASE alignment of structures representing prolyl- and threonyl-tRNA synthetases are shown in Figure 6.4A. Visual inspection of the structure alignment suggests the region is well-aligned. However, Figure 6.4B shows an alternative alignment with lower local covariation. The structure alignments shown in Figure 6.4A and Figure 6.4B appear to be of equivalent quality when visually inspected. However, the realigned structures in Figure 6.4B show an improvement to the RMSD scores. The RMSD of the orthologous structures, 1H4Q and 1NJ8, improves from 2.44Å to 0.73Å . The RMSD of the paralogous structures, 1H4Q and 1EVK, improves from 2.85 Å to 1.67 Å. Thus when aligning divergent structures, misalignments may be undetectable by visual inspection.

In Figure 6.4C and D, we analyze only the orthologous sub-family to clarify the structure

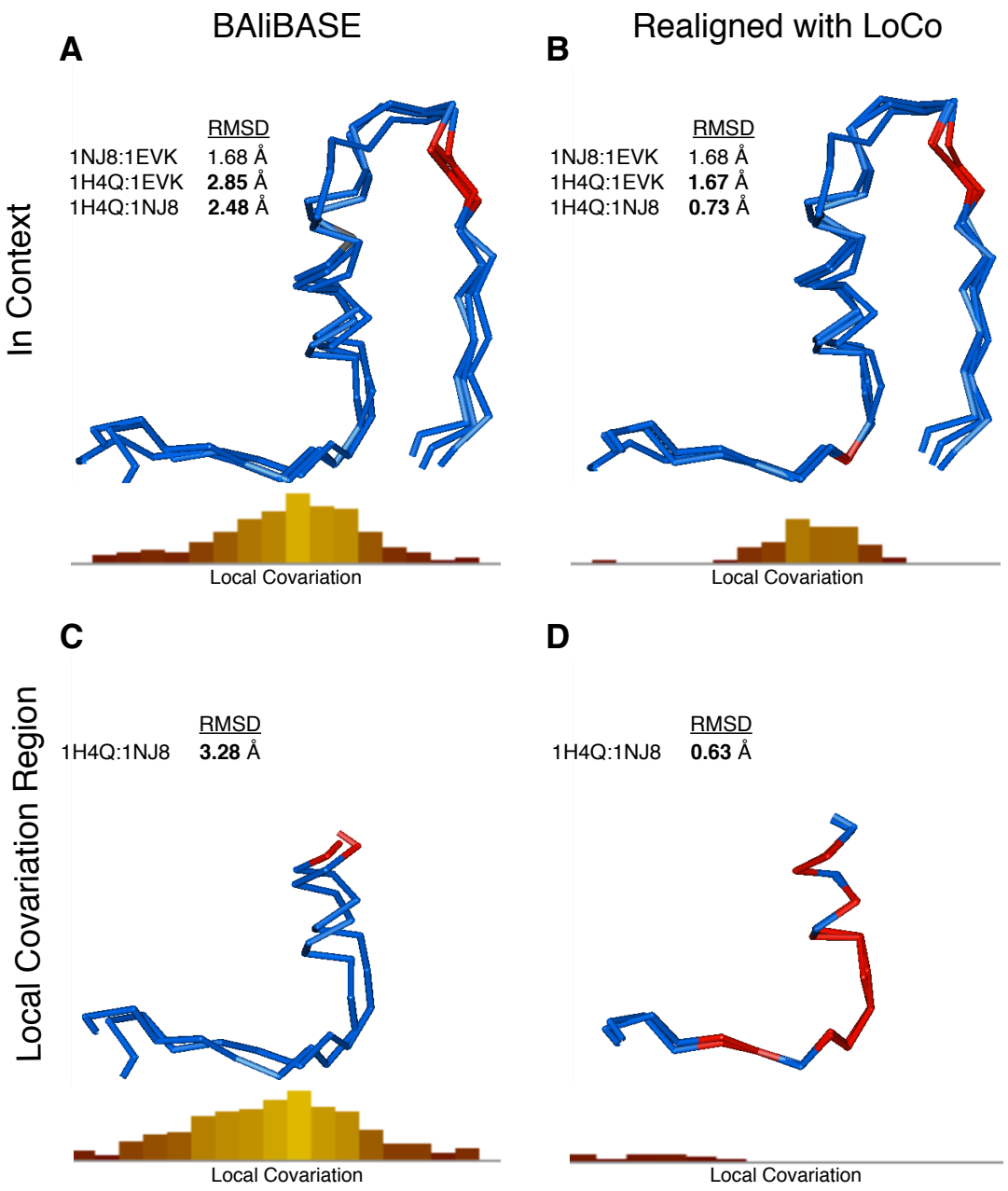

Figure 6.4: Local covariation identifies structural alignment error in BAliBASE 3 alignment of tRNA Synthetases (BB30002). Each panel shows a structure alignment built with Cn3D [18] with the corresponding local covariation histogram from LoCo below. (**A**) Structure alignment of the tRNA synthetase subfamilies from BAliBASE 3. Structures are coloured by fit and the maximum local covariation value (2.8) implies a misalignment exists. (**B**) Realignment of misaligned structure from panel **A** reduces local covariation (maximum peak 1.8). Both panels **A** and **B** look very similar which explains why misalignment was missed during BAliBASE manual curation process. (**C**) Structure alignment of *only* the misaligned region of Prolyl tRNA Synthetase subfamily from panel **A**. Structures are coloured by identity such that conserved residues are red. Local covariation maximum is 3.0. (**D**) Realignment of panel **C** to minimize local covariation. Minimizing local covariation produces marked improvement in both the structure alignment quality and sequence conservation.

misalignment visually. Figure 6.4C shows the alignment of the high local covariation region of only the prolyl-tRNA synthetase subfamily of BB40002. This region shows poor structural conservation and residue identity. When the shift error is resolved using LoCo as a guide, the quality of the alignment is markedly improved (Figure 6.4D). The alignment shows improved sequence conservation and a much lower RMSD, from 3.28 Å to 0.63 Å. The local covariation present in Figure 6.4C is no longer present in Figure 6.4D.

The BB30 category of alignments are designed to test the ability to properly align multiple subfamilies into a single subalignment. Aligning paralogous sequences is particularly challenging because of increased sequence divergence and different functional constraints. Functional divergence can result in increased substitution rates (type I divergence) [15]. Divergence can also occur without a change in substitution rate in the form of differing residue properties allowed at a given position (type II divergence) [16, 17]. These types of divergence can make it difficult to determine the alignment between paralogous proteins from sequence alone. However, the misalignment presented in BB30002 is within a subfamily and is between two structures. The discovery of a structural misalignment between two similar sequences from the same subfamily in a hand-curated alignment demonstrates the importance of independent validation of sequence and structure alignments.

6.3.5 Local Covariation Identifies Active Site Residues

Not all regions of high local covariation in BAliBASE are explained by potential misalignments; in fact, some segments with high local covariation are structurally valid. It is thus crucial to examine regions of high local covariation to determine the root cause. As outlined in this section, local covariation can identify segments of interest that covary because of another mechanism. In our analysis of BAliBASE 3, we identified BB11003 as an alignment with a region of high local covariation(Figure 6.5). Two structures, 1AD3 and 1EYY, are of aldehyde dehydrogenase; the other structures are of carboxylate dehydrogenase (1UZB), and γ-glutamyl phosphate reductase (1O20). We investigated this alignment for an explanation of the high

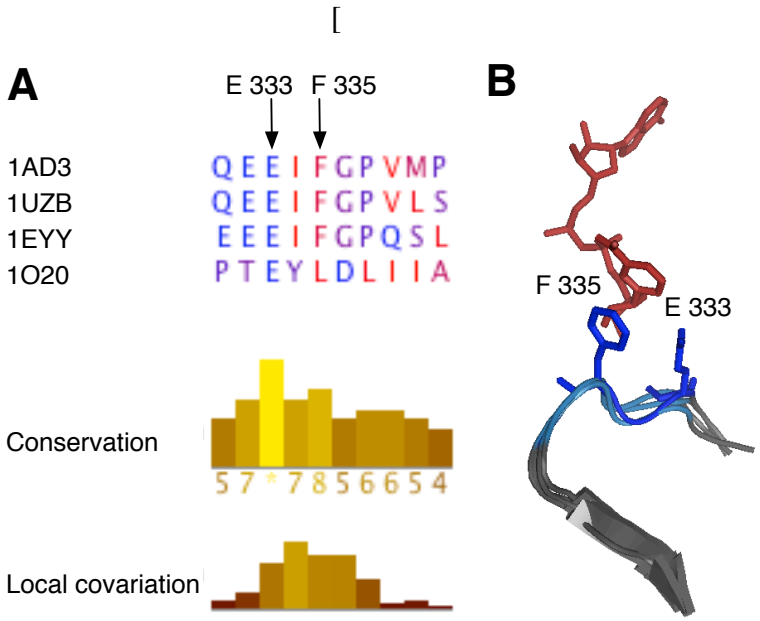

Figure 6.5: Local covariation identifies active site residues. (**A**) Screenshot from the LoCo tool showing the region of high local covariation from BAliBASE 3 alignment BB11003. BB11003 is an alignment of four paralogous oxireductases with similar structure. The local covariation peaks highlight four positions in the sequence alignment which are coloured in blue in panel **B**. Two active site residues from structure 1AD3, E333 and F335, are emphasized in the sequence alignment. (**B**) Structure alignment of residues shown in panel **A** made in PyMOL [5]. The region of high local covariation is highlighted in blue; structure 1AD3 is emphasized with dark blue. The NAD cofactor from structure 1AD3 is drawn in red. Important binding residues E333 and F335 from 1AD3 are rendered in sticks representation.

local covariation. The sequence alignment in the region of high local covariation (Figure 6.5A) is supported by the structure alignment in the same region (Figure 6.5B). Thus, we concluded that shift error did not explain the high local covariation.

As noted in the previous section, protein families undergo functional divergence after gene duplication leading to paralogous alignments have specific characteristics. Functional divergence that occurs at a only a clustered subset of positions will cause an increase in local covariation. Thus, the presence of paralogues that have undergone type II divergence [16, 17] may create a false-positive detection of misalignment. However, we show that the detection of type II divergence in important functional regions may prove useful for identifying binding sites or understanding divergence in paralogous families.

Structure 1AD3 included the coenzyme NAD, which is critical for enzyme function. Figure 6.5B shows that the region of high local covariation (blue) is oriented towards NAD (red). The region of high local covariation spans four residues: E333, I334, F335, and G336. E333 is absolutely conserved and therefore cannot contribute to any covariation score. The other three residues, I334, F335, and G336, vary in the sequence of 1O20 but not in the backbone structure. Because NAD is critical for catalysis [31], we hypothesized that the contacts made by E333 and F335 could be important for function [27].

The human homologues for positions E333 and F335, E399 and F401 respectively, have been found to be important for function. The human E399 binds the NAD ribose; mutations to the position significantly affect the catalytic rate [29]. F335 orients NAD through an aromatic stacking interaction Figure [27]. Thus, local covariation identified important functional residues from a paralogous protein family. This example illustrates that not all regions of high local covariation are caused by misalignments. Thus, it is important to visually inspect regions of high local covariation to elucidate the cause.

6.4 Discussion

Protein family misalignments can cause errors in downstream analyses — unimportant positions may be falsely identified as conserved or coevolving and critical conserved positions can be overlooked. Systematic misalignments can reduce the bootstrap values of phylogenetic trees or reinforce incorrect trees [25]. Thus, it is critical that alignments be validated by a criterion independent of the assumptions used to build them.

Selecting which alignment is most likely correct can be a source of debate because there is no high-throughput biochemical method to prove the validity of an alignment. Some investigators prefer to believe the internally consistent output of an established alignment algorithm over an alternative alignment with some biological justification. Here we provide a tool to identify regions in an alignment that should be investigated. Automating alignment using local covariation as a parameter is difficult because increased local covariation is not tautologically equivalent to misalignment, as shown by the example of correctly aligned paralogs in Figure 6.5. However, as a guide for curation of protein alignments, the tool is extremely effective at identifying regions of potential misalignment [14, 23, 6, 34].

We provide strong structural evidence of the validity of our alternative alignments over the BAliBASE alignments in the form of cysteine conservation at a disulphide bond (Figure 6.3) and significantly improved RMSD of a structure alignment (Figure 6.4). As noted by Kuziemko et al., the alignment supported by structural evidence may receive a lower score than an alignment which simply optimizes the sequence alignment algorithm's scoring function [24]. This observation suggests that we should be skeptical of alignments that are validated only by an alignment scoring function. Furthermore, the existence of potential misalignments in the most widely used, hand-curated benchmark dataset implies that such misalignments may be common in high-throughput datasets of lower quality.

Large datasets are known to have many systematic misalignments caused by incorrect sequential or structural inference because of the limitations of current alignment methods [9, 22]. Many alternative alignments may seem equally valid because there are no methods to prove

the correct alignment aside from solving the structures for all proteins in the alignment. Thus, identification of serious errors with significant contradictory structural evidence is a method for demonstrating an alignment is incorrect. Such structurally corroborated misalignments are rare, especially in curated datasets. Nevertheless, the misalignments we identified in Figure 6.3 and Figure 6.4 provide such structural evidence.

It is interesting to contrast this assessment of BAliBASE 3 with a previous analysis of BAliBASE by Edgar [9]. Both studies investigate the quality of alignment benchmarks using criteria independent of sequence conservation. The different criteria for evaluating BAliBASE highlighted different sets of BAliBASE alignments for discussion. Edgar used domain homology and secondary structure annotations to assess alignment quality; he argues correctly that alignments of sequences with conflicting annotations are less reliable for benchmarking. In this manuscript, we identify structurally supported shift errors in the same dataset and, by extension, other similar datasets. These two studies form complementary assessments of the BAliBASE benchmark set.

The exploration of the BAliBASE BB30 subfamilies dataset, as in Figure 6.4, draws attention to the concept of homology and sequence alignment. Alignments designed to search for coevolving positions in a protein should ideally be orthologous, comprising sequences related by linear descent. However, sequences can also be homologous (similar by common evolutionary history) because of paralogy (related through a gene duplication event). Paralogous positions may be under different functional constraints [15, 16, 17]; an example would be the tRNA synthetases shown in Figure 6.4A and B. While both subfamilies are tRNA synthetases, they catalyze a reaction with different tRNAs and different amino acids. Although more exploration is needed, the inclusion of paralogous sequences could potentially increase local covariation to a lesser extent than misalignments. The presence of paralogous sequences may explain the occurrence of covariation within binding sites.

Identifying functional residues is an important open problem. The degree of conservation of a position is typically used to indicate its potential importance in such analyses. However,

when paralogous families are included and conservation is lost, local covariation could also be used to search for non-conserved, functionally important residues. We provide an example of local covariation in a functional region in our analysis of the alignment BB11003. Investigating the region of local covariation revealed two important functional residues in an alignment of 4 sequences. Residues E333 and F335 both make important contacts to NAD in the coenzyme binding site (Figure 6.5).

Local covariation previously identified an interesting structural region in phosphoglycerate kinase [14]. In this example, a linker region contained either a sheet or a helix to serve the same structural purpose. Technically, the region was not shifted because there was no alternative alignment; there was simply no structurally meaningful alignment between the two sequence subsets. These examples illustrate that it is critical that alignments be visually inspected regardless of the method used to generate them.

An interesting illustration of the importance of manual alignment curation is provided by Kawrykow et al. through their work on the sequence alignment game Phylo [21]. Phylo uses the concept of crowdsourcing to improve sequence alignments by having human players inspect and correct them. It is important to note that Kawrykow et al. found that untrained game players were able to outperform the top performing automated solutions. This observation reinforces the importance of visually inspecting alignments after they are built by an automated solution; LoCo provides an interface to guide and expedite the investigation.

Increased local covariation should not be confused with patch covariation, where two short contiguous segments of sequence coevolve with one another [39]. Increased local covariation is only concerned with covariation that occurs within a short segment of an alignment, not between segments. As we noted previously, it is possible to use covariation statistics like Zpx and ΔZp to find true coevolving pairs that are distant in sequence even in regions of misalignment [6].

We have made the tool used in this manuscript, LoCo, available online. LoCo can be used effectively on large datasets. Performance can become a concern when analyzing alignments

with many ungapped positions because of the covariation calculations. However, because the covariation algorithms are implemented in C and optimized, we have successfully analyzed very large concatenated protein datasets with thousands of sequences. We have run LoCo successfully on concatenated alignments over 2500 ungapped positions long, though at this size the covariation module requires approximately 1 gigabyte of memory and 1 minute of CPU time to update the local covariation score. Alignments this size can be analyzed because of the extensive optimizations made to the covariation calculation software. LoCo and its antecedents have been an important part of building high quality protein alignments for several recent manuscripts [14, 23, 6, 34]. Using LoCo, we have seen marked improvement in our sequence alignment quality, confidence, and downstream analyses.

Analyses of alignments which contain errors are inherently unreliable. LoCo provides an intuitive and rapid platform to identify and correct alignment errors. We recommend that new alignments be analyzed with local covariation and visually inspected before any conclusions are drawn from them.

6.5 Materials and Methods

6.5.1 Demonstrating Local Covariation Rationale

We created a 7-position synthetic alignment to demonstrate the effectiveness of local covariation for finding misalignments (Figure 6.1). Each column in the misalignment contained a randomly assorted subset of 3 residues that was mutually exclusive with adjacent columns; this alignment was called ‘No Shift’. The ‘3% Alignment Shift’ and ‘5% Alignment Shift’ alignments were created by randomly shifting a subset of sequences one position to the right, 6 of 200 and 10 of 200, respectively. Figure 6.1A shows the shift of positions 2–6 diagrammatically. Positions and 1 and 7, which flank the misaligned region, remain unshifted.

The synthetic alignments were inserted at the N-terminus of a structure-guided and manually-curated alignment of methionine aminopeptidase. We subsequently analyzed the synthetic

alignment using the covariation statistic *Zp* [7, 6] Conservation scores were calculated using Jalview [38].

6.5.2 Algorithm Overview

LoCo calculates the average covariation between positions in a protein alignment using the *Zp/MIp* statistic [7] using a compiled program written in C. The algorithm for calculating *Zp* is optimized for memory use and speed. *Zp* is based on mutual information, a statistic that is calculated based on the relative counts and pairwise counts of each individual alignment position.

MIp is defined as:

$$MIp_{i,j} = MI_{i,j} - (\overline{MI}_{i,x} \times \overline{MI}_{j,x})/\overline{MI} \tag{6.1}$$

where $\overline{MI}_{i,x}$ is the mean Mutual Information of position i with all other positions and $\overline{MI}$ is the overall mean Mutual Information. *MIp* is normalized and referred to as *Zp*:

$$Zp_{i,j} = (MIp_{i,j} - \overline{MIp})/\sigma(MIp) \tag{6.2}$$

where again $\overline{MIp}$ is the mean *MIp* and $\sigma(MIp)$ is its standard deviation. The convention of referring to normalized *MIp* as *Zp* was introduced in [6].

Because there are 20 amino acids, there are 20 potential entries in the count matrix; each pairwise count represents two positions so there are 400 potential entries for each pairwise count. However, because the majority of positions demonstrate some degree of conservation, most entries in the count and pairwise count matrices will be zero. This fact is exploited by the LoCo algorithm — a reusable linear array is used to initialize a dynamically allocated linked list which stores the pairwise count for each pair of positions for significant memory savings.

Local covariation is calculated by taking the average *Zp* score between all pairs of positions over a window of six; this is done in a Perl script upon completion of the C program.

The programs used to calculate covariation statistics can be used independently of the Jalview GUI. These programs are accessed using the Perl script MIp.pl; they take a fasta-formatted alignment and, optionally, a pdb-formatted structure as input and return a summary file of covariation statistics (and inter-residue distances if the pdb file is provided). The MIp software can be automated to screen large alignment datasets.

6.5.3 The LoCo Alignment Curation Tool

The alignment editing software is a modified version of Jalview [38]. Because covariation statistics can be time-consuming to calculate, the major calculations are computed using an optimized algorithm implemented in the C programming language. The default Jalview sequence alignment window displays protein sequences above three indicators of alignment quality — Conservation, Quality score and Consensus. Because quality scores are based on conservation, in LoCo we have replaced Quality with Local Covariation. High local covariation indicates a high likelihood of systematic misalignment in that region, regardless of conservation score.

6.5.4 The LoCo Alignment Curation Procedure

We have developed a simple procedure to correct potential systematic misalignments using LoCo: 1) Identify potential misalignments (Figure 6.3A), 2) cluster using neighbour joining by percent identity (Figure 6.3D), 3) test alternate alignments (Figure 6.3B).

Potentially misaligned regions can be identified by examining the ”Local Covariation” bar at the bottom of the alignment window. In [6], we noted that a local covariation score above 2.5 was worth investigating; however, we have found that cutoff to be conservative. Covariation scores are affected by the number of sequences in the alignment and by their similarity, so it is possible to find misalignments in small alignments (approximately 10 sequences) with much lower local covariation scores. Alignments with fewer sequences have narrower distributions of covariation. We recommend investigating any position where the local covariation score 1) appears to be above the ‘background’ for the alignment, 2) is increased for several adjacent

positions, or 3) is above 2.0 (coloured yellow in the histogram).

Clustering is done by highlighting the potentially misaligned positions and selecting "Neighbour Joining Using % Identity" from the Calculate menu. Regions of systematic misalignment will cluster separately from correctly aligned sequences. Sequences can be placed in the same order as the tree by using the Sort command in the Calculate menu.

Finally, alternate alignments can be tested by highlighting the region of misalignment and dragging the misaligned sequence into position by holding control while left-clicking and dragging the mouse. The local covariation score will change as you edit the alignment.

6.5.5 Automated Search of CDD and BAliBASE

We collected alignments from the Conserved Domain Database [28] from

ftp://ftp.ncbi.nih.gov/pub/mmdb/cdd/

We collected all sequences from the ftp distribution of BAliBASE 3 [36] from

ftp://ftp-igbmc.u-strasbg.fr/pub/BAliBASE3/

The BAliBASE alignments were converted to fasta format by readseq [13]. A simple Perl-based pipeline was used to automate the use of the MIp.c and MIp.pl programs used to calculate covariation in the LoCo alignment curation tool. We counted the number of local covariation peaks at or above the 2.0 threshold considered worth investigating. The number of peaks above 2.0 were plotted in R [19]; contiguous blocks were coloured as they represented an extended region of high local covariation.

6.5.6 Structure Validation

Structures were collected from the RCSB Protein Data Bank [3]. Structure alignments for Figure 6.3 and Figure 6.4 were made using Cn3D [18]. The Cn3D alignments are coloured by identity such that conserved positions are coloured red and non-conserved positions are coloured blue. RMSD for structure alignments was calculated using PyMOL [5]. The structure alignment for Figure 6.5 was created using PyMOL[5]. The entire structure alignment

was rendered using the ‘cartoon’ renderer. Important residues and the NAD cofactor are emphasized through stick rendering on top of the original alignment. NAD is coloured red. The region of high local covariation is coloured blue.

Bibliography

[1] W R Atchley, K R Wollenberg, W M Fitch, W Terhalle, and A W Dress. Correlations among amino acid sites in bhlh protein domains: an information theoretic analysis. *Mol Biol Evol*, 17(1):164–178, 2000.

[2] WR Atchley, KR Wollenberg, WM Fitch, W Terhalle, and AW Dress. Correlations among amino acid sites in bhlh protein domains: an information theoretic analysis. *Molecular Biology and Evolution*, 17(1):164, 2000.

[3] H.M Berman, J Westbrook, Z Feng, G Gilliland, TN Bhat, H Weissig, I.N Shindyalov, and P.E Bourne. The protein data bank. *Nucleic Acids Research*, 28(1):235, 2000.

[4] Michele Clamp, James Cuff, Stephen M Searle, and Geoffrey J Barton. The jalview java alignment editor. *Bioinformatics*, 20(3):426–7, Feb 2004.

[5] WL Delano. The pymol molecular graphics system. Jan 2002.

[6] R.J Dickson, L.M Wahl, A.D Fernandes, and G.B Gloor. Identifying and seeing beyond multiple sequence alignment errors using intra-molecular protein covariation. *PLoS One*, 5(6):e11082, 2010.

[7] SD Dunn, LM Wahl, and GB Gloor. Mutual information without the influence of phylogeny or entropy dramatically improves residue contact prediction. *Bioinformatics*, 23(3):333–340, 2008.

[8] SD Dunn, LM Wahl, and GB Gloor. Mutual information without the influence of phylogeny or entropy dramatically improves residue contact prediction. *Bioinformatics*, 24(3):333, 2008.

[9] R.C Edgar. Quality measures for protein alignment benchmarks. *Nucleic Acids Research*, 38(7):2145, 2010.

[10] M.A Fares and S.A.A Travers. A novel method for detecting intramolecular coevolution: adding a further dimension to selective constraints analyses. *Genetics*, 173(1):9, 2006.

[11] J Felsenstein. Inferring phylogenies. *Sunderland*, Jan 2004.

[12] WM Fitch and E Markowitz. An improved method for determining codon variability in a gene and its application to the rate of fixation of mutations in evolution. *Biochemical Genetics*, 4(5):579–593, 1970.

[13] Don Gilbert. Sequence file format conversion with command-line readseq. Feb 2003.

[14] Gregory B Gloor, Gaurav Tyagi, Dana M Abrassart, Andrew J Kingston, Andrew D Fernandes, Stanley D Dunn, and Christopher J Brandl. Functionally compensating coevolving positions are neither homoplasic nor conserved in clades. *Mol Biol Evol*, 27(5):1181–91, May 2010.

[15] X Gu. Statistical methods for testing functional divergence after gene duplication. *Mol Biol Evol*, 16(12):1664–74, Dec 1999.

[16] X Gu. Maximum-likelihood approach for gene family evolution under functional divergence. *Mol Biol Evol*, 18(4):453–64, Apr 2001.

[17] Xun Gu. A simple statistical method for estimating type-ii (cluster-specific) functional divergence of protein sequences. *Mol Biol Evol*, 23(10):1937–45, Oct 2006.

[18] C W Hogue. Cn3d: a new generation of three-dimensional molecular structure viewer. *Trends Biochem Sci*, 22(8):314–6, Aug 1997.

[19] R Ihaka and R Gentleman. R: a language for data analysis and graphics. *Journal of computational and graphical statistics*, pages 299–314, 1996.

[20] I Kass and A Horovitz. Mapping pathways of allosteric communication in groel by analysis of correlated mutations. *Proteins*, 48(4):611–617, 2002.

[21] Alexander Kawrykow, Gary Roumanis, Alfred Kam, Daniel Kwak, Clarence Leung, Chu Wu, Eleyine Zarour, Phylo players, Luis Sarmenta, Mathieu Blanchette, and Jérôme Waldispühl. Phylo: a citizen science approach for improving multiple sequence alignment. *PLoS One*, 7(3):e31362, 2012.

[22] Changhoon Kim and Byungkook Lee. Accuracy of structure-based sequence alignment of automatic methods. *BMC bioinformatics*, 8:355, 2007.

[23] B. P Kleinstiver, A. D Fernandes, G. B Gloor, and D. R Edgell. A unified genetic, computational and experimental framework identifies functionally relevant residues of the homing endonuclease i-bmoi. *Nucleic Acids Research*, 38(7):2411–2427, Apr 2010.

[24] Andrew Kuziemko, Barry Honig, and Donald Petrey. Using structure to explore the sequence alignment space of remote homologs. *PLoS Computational Biology*, 7(10):e1002175, 2011.

[25] J.A Lake. Reconstructing evolutionary trees from dna and protein sequences: paralinear distances. *Proceedings of the National Academy of Sciences*, 91(4):1455, 1994.

[26] D.Y Little and L Chen. Identification of coevolving residues and coevolution potentials emphasizing structure, bond formation and catalytic coordination in protein evolution. *PLoS One*, 4(3):e4762, 2009.

[27] Z.J Liu, Y.J Sun, J Rose, Y.J Chung, C.D Hsiao, W.R Chang, I Kuo, J Perozich, R Lindahl, and J Hempel. The first structure of an aldehyde dehydrogenase reveals novel interactions

between nad and the rossmann fold. *Nature Structural & Molecular Biology*, 4(4):317–326, 1997.

[28] A Marchler-Bauer, AR Panchenko, BA Shoemaker, PA Thiessen, LY Geer, and SH Bryant. Cdd: a database of conserved domain alignments with links to domain three-dimensional structure. *Nucleic Acids Research*, 30(1):281, 2002.

[29] L Ni, S Sheikh, and H Weiner. Involvement of glutamate 399 and lysine 192 in the mechanism of human liver mitochondrial aldehyde dehydrogenase. *Journal of Biological Chemistry*, 272(30):18823, 1997.

[30] O Olmea, B Rost, and A Valencia. Effective use of sequence correlation and conservation in fold recognition1. *Journal of molecular biology*, 293(5):1221–1239, 1999.

[31] S.J Perez-Miller and T.D Hurley. Coenzyme isomerization is integral to catalysis in aldehyde dehydrogenase. *Biochemistry*, 42(23):7100–7109, 2003.

[32] Art Poon and Lin Chao. The rate of compensatory mutation in the dna bacteriophage phix174. *Genetics*, 170(3):989–999, 2005.

[33] A Rodionov, A Bezginov, J Rose, and E.R.M Tillier. A new, fast algorithm for detecting protein coevolution using maximum compatible cliques. *Algorithms for molecular biology*, 6(1):17, 2011.

[34] Ryo Takeuchi, Abigail R Lambert, Amanda Nga-Sze Mak, Kyle Jacoby, Russell J Dickson, Gregory B Gloor, Andrew M Scharenberg, David R Edgell, and Barry L Stoddard. Tapping natural reservoirs of homing endonucleases for targeted gene modification. *Proc Natl Acad Sci U S A*, 108(32):13077–82, Aug 2011.

[35] R Thangudu, M Manoharan, N Srinivasan, F Cadet, R Sowdhamini, and B Offmann. Analysis on conservation of disulphide bonds and their structural features in homologous protein domain families. *BMC Structural Biology*, 8(1):55, 2008.

[36] J.D Thompson, P Koehl, R Ripp, and O Poch. Balibase 3.0: latest developments of the multiple sequence alignment benchmark. *Proteins*, 61(1):127–136, 2005.

[37] JD Thompson, F Plewniak, and O Poch. Balibase: a benchmark alignment database for the evaluation of multiple alignment programs. *Bioinformatics*, 15(1):87, 1999.

[38] A.M Waterhouse, J.B Procter, D Martin, M Clamp, and G.J Barton. Jalview version 2—a multiple sequence alignment editor and analysis workbench. *Bioinformatics*, 25(9):1189, 2009.

[39] Y.B Xu and E.R.M Tillier. Regional covariation and its application for predicting protein contact patches. *Proteins*, 78(3):548–558, 2010.

[40] C Yanofsky, V Horn, and D Thorpe. Protein structure relationships revealed by mutational analysis. *Science*, 146(3651):1593, 1964.

Chapter 7

Discussion

This work represents a contribution to our collective understanding of sequence alignment, coevolution, and their intertwined relationship. Prior to this work, the alignment-coevolution relationship was not adequately characterized in the literature, which has led to misuses of coevolutionary methods. This work is an attempt to acknowledge the fundamental implicit assumptions made about evolution when conducting sequence analysis experiments *in silico*. It is also an advancement of the fields of multiple sequence alignment, and (both directly and indirectly) coevolutionary inference.

7.1 Improvements to multiple sequence alignment

7.1.1 Putative misalignments in benchmark alignment databases

Multiple sequence alignment saw a decade of advancement after the publication of the BAliBASE benchmark database [26] because it allowed computationally-focused researchers to treat sequence alignment as an optimization problem, and it allowed the direct competition of various methods with evidence to determine which was best and why. This environment facilitates innovation and advancement. But after a decade of impressive innovation, the field stagnated because new techniques were not able to outperform established methods on stan-

dard benchmarks. It appeared that the maximum possible accuracy had been reached and that it was less than 100%.

However, the local covariation analysis (described in chapter 6) of the most current BAliBASE has revealed that the benchmark database contains alignment errors. Interestingly, the errors found using local covariation were of a different type compared to previous analyses [9], and thus the errors discussed in Chapter 6 complement the analysis by Edgar. The presence of errors implies that the reason new methods could not improve upon established ones is that researchers were trying to optimize their tools for contradictory data. Researchers were trying to optimize their tools to match BAliBASE alignments, and, with no optimal solution, created tools that generated systematic errors elsewhere. Thus, the limits of multiple sequence alignment accuracy were not reached, but rather, the limits of the accuracy of the benchmark were reached.

7.1.2 Alignment reliance on conservation

Local covariation and LoCo were able to identify alignment errors because local covariation is an independent criterion on which an alignment can be validated [6]. Multiple sequence alignment algorithms exist to maximize conservation, sequence identity and similarity defined by the scoring matrix used. Iterative methods test alternative alignments to prevent systematic errors, but still use conservation as the criterion by which to identify misalignment [10, 17]. Researchers then analyze alignments to find conserved positions.

The fact that this process is based entirely on conservation means that errors that result in some erroneous, though highly-conserved positions, will be incorrectly inferred as important to structure and function. This wastes valuable experimental resources. And because the errors are conservation-based, it is unlikely that conservation-based curation methods will detect them all.

Alignment databases like BAliBASE attempt to circumvent this problem by using structure as the independent criterion on which to validate the sequence alignment. However, this is not

an adequate solution because structure is comparatively rare, and because, as demonstrated in chapter 6, alignment that contradicts the structural model can still occur because of structure alignment shift errors or human error. Furthermore, the intention of using high quality multiple sequence alignments for *de novo* structure prediction means that a reliance on structure for validation is not possible.

7.1.3 Supporting alternative alignments

One of the biggest challenges associated with the analysis of a benchmark database is the presumption of correctness. A great deal of literature relies on the absolute correctness of a benchmark database, and thus work that is critical of it can be a lightning rod for controversy [9]. Chapter 6 identifies many potential misalignments in major databases, but it is difficult to prove conclusively that an error has occurred in many cases; the burden of proof lies with the accuser.

The potentially-misaligned proteins in high quality datasets often look correct when using traditional tools, or else they would not have been included in such a dataset. In the typical case, when two equally-plausible alignments exist, the default is to trust the database. When local covariation suggests an alternative alignment, it is met with skepticism. This is why, of the many potential misalignments which were identified in BAliBASE, a specific three were chosen for demonstration; each of these alignments has independent structural or phylogenetic information on which the correct alignment can be checked. Thus, the violations of the respective structural models provides compelling evidence for the correctness of the alternative alignment suggested by local covariation.

7.1.4 Local covariation is not dependent on conservation

Local covariation provides an independent validation of sequence alignment correctness [5, 6]. Sequence alignments can be built using conservation and then checked for systematic errors using local covariation. Presently, local covariation is a heuristic and not a metric, and thus it

cannot be used to automate alignment curation.

When a systematic misalignment is repaired, the local covariation of the misaligned region will be reduced. However, the correct alignment is not the only potential arrangement which will yield low local covariation. If sequences were arranged randomly, there would be no sequence-local positional correlations either. Of course, such a random assortment of sequences would have low conservation and would be identified as potentially-misaligned through the conservation metric. In this way, local covariation functions similarly to an alignment position validation tool like GBLOCKS [25].

GBLOCKS is an important part of a phylogenetic inference pipeline because it removes positions from an alignment which are not of sufficient quality to be included in the phylogenetic algorithm, which typically includes positions with gaps and positions with low conservation [25]. But because GBLOCKS is based on conservation, it may infer misalignments of analogous residues as being correctly aligned. Local covariation can provide a similar function to GBLOCKS by identifying potentially misaligned regions which should be removed prior to phylogenetic analysis.

7.1.5 Local covariation as an alternative to GBLOCKS

In [24], I created a multiple sequence alignment of the LAGLIDADG Homing Endonuclease protein family with the intention of showing the diversity of the protein family that could be exploited for enzyme design purposes. The diversity of the LAGLIDADG family makes sequence alignment very challenging [3]. The LAGLIDADG family targets diverse binding sites and the amino acid diversity seen at binding locations is characterized as an example of "rapid evolution" [19].

Furthermore, the LAGLIDADG homing endonuclease is a two-domain enzyme which exists in either a homodimer or single chain form, each of which is separated into a different protein subfamily in the PFAM database [23]. Misclassification of single chain LAGLIDADG enzymes as the homodimer form and vice versa is a serious problem in the creation of a LAGL-

IDADG protein family. The single chain form of the LAGLIDADG enzyme is connected by a long, variable, unstructured, linker region that, coupled with low sequence identiy, makes automated alignment of the family very challenging.

GBLOCKS [25] was unsuccessful in selecting positions for phylogenetic analysis because GBLOCKS relies on conservation to indicate which positions are well-aligned and the highly divergent LAGLIDADG family was too diverse. LoCo [6] was used to curate the LAGLIDADG protein family instead. Local covariation highlighted regions of potential misalignment which were either corrected if possible, or removed from the phylogenetic analysis. Regions of high local covariation with no obvious alternative alignment were removed from the phylogenetic analysis. In this way, local covariation was used in place of conservation as an independent tool for validating an alignment for phylogenetic analysis. The result was a phylogenetic tree with high confidence that demonstrated the enormous diversity of the LAGLIDADG family.

7.2 Improvements to structure prediction and coevolutionary inference

7.2.1 Sequence alignment improvement critical to application of coevolutionary methods

The first observation that I made regarding the *MIp* [8] covariation statistic was that its impressive contact prediction accuracy was not consistent when applied to so-called "gold standard" public databases. When analyzing the highest quality alignments, which are structure-guided and painstakingly hand-curated, *MIp* produced impressive residue contact prediction accuracy.

However, analyzing alignments in publicly-available databases produced inconsistent results, even when controlling for sequence diversity, length, and number. It was clear that this tool for coevolution inference would not become a widely adopted tool for protein family anal-

ysis if one could not determine whether its predictions were meaningful. This is especially true because the accuracy of coevolution predictions is checked against the structure of a member of the protein family; such information would certainly not be available when using coevolution to contribute to the *a priori* structure prediction problem.

7.2.2 Misalignment as a source of false-positive covariation

Atchley et al. had suggested that the covariation between two positions in a protein family could be broken down into a linear combination of several components [1]:

$$C_{ij} = C_{phylogeny} + C_{structure} + C_{function} + C_{interactions} + C_{stochastic}$$

The *APC* correction applied to *MI* to create *MIp* had removed the phylogenetic noise component from such a breakdown, but it was possible that other sources of noise contaminated the coevolution signal. Based on the results discussed in Chapter 5, I now believe that systematic misalignments represent another component of the linear combination of factors that contribute to coevolution signal, previously hidden within the $C_{stochastic}$ component:

$$C_{ij} = C_{phylogeny} + C_{structure} + C_{function} + C_{interactions} + C_{stochastic} + \mathbf{C_{systematic\_misalignments}}$$

This result was very counter-intuitive. It was expected that misalignments would obscure coevolution signal and result in reduced covariation scores. In fact, anecdotally, it was hypothesized that alignments could be improved by maximizing coevolution, since that was believed to be a measure of the quality of an alignment.

7.2.3 Development of new coevolution-inferring statistics

Another key observation was that the coevolution signal was still present in the correctly-aligned regions of an alignments that contain some systematic misalignments. The simplest and most common practice for evaluating coevolution predictions was to evaluate the top-scoring N pairs where N was either some fixed number, or some function of the length of the protein. The fact that systematically-misaligned regions dramatically increased covariation scores meant that the other predictions for the well-aligned regions were overlooked.

The presence of overshadowed coevolution predictions in misaligned proteins led to the development of two new coevolution-predicting statistics, Zpx and ΔZp [5], introduced in Chapter 5. Both of these statistics are designed to find outliers from each respective position's distribution of coevolution scores, rather than the distribution of all pairs. These new statistics improve contact prediction in optimal alignments. But they are especially good at identifying contacting positions in alignments which contain systematic misalignments, compared to earlier coevolution-inference methods.

It is interesting to note that each statistic finds a different subset of all putatively coevolving pairs of positions for a protein family, but the consensus set agreed upon by MIp, Zpx, and ΔZp are the most likely to be in contact. Thus, advances in sequence alignment and advances in coevolution inference have created a system of tools for coevolution inference that are more generalizable and more accurate.

7.2.4 Gaps are not the 21st amino acid

Finally, it is critically important that gaps are treated correctly throughout coevolutionary inference. As demonstrated in Chapter 4, gap characters create the illusion of covariation when treated as the 21st character in an alignment. It is very concerning that Mutual Information-based methods like MIp have been modified to include gaps as the 21st character, with no acknowledgement to the significance of such a modification. Furthermore, the spurious coevolution signal seen from such positions has been interpreted as being functionally important [18].

Furthermore, Chapter 4 demonstrated that including gaps as the 21st character decreases contact prediction accuracy. It is a common practice to benchmark against established tools to assess a new method's utility. Some recent novel methods have been benchmarked against Mutual Information-based statistics that were modified to include gaps as the 21st character [21]. The modification of the existing tool invalidates such a benchmark.

7.3 Validating coevolution predictions

7.3.1 Active site variation constraints from a coevolving network of positions

Atchley's linear combination model of covariation signal highlights the existence of functional-based covariation. The possibility of functional coevolution was explored using the LAGLIDADG HE family initially investigated in Takeuchi et al. [24]. An analysis of the LAGLIDADG HE family by *Zpx* revealed that the pair with the strongest coevolution signal was between the N- and C-terminal domains, across the central helix, and adjacent to the two respective catalytic residues on the N-terminal side.

These residues formed a network of coevolving residues which included a catalytic residue; this is very rare, because catalytic residues are typically conserved and thus cannot covary. This work, described by McMurrough et al. (manuscript in prep) [20] establishes that there is a coevolving network of residues which controls the variation of the active site. The discovery of this coevolving network is of high utility for enzyme design work, as it can potentially guide the necessary sequence changes necessary to retain the function of a designed enzyme.

7.3.2 Validation of consensus coevolution predictions in phosphoglycerate kinase

Another validation of the predictions of *MIp* [8], *Zpx*, and ΔZp [5] was completed by Gloor et al. [13]. In this work, Gloor et al. applied local covariation and the aforementioned coevolutionary-inference methods to phosphoglycerate kinase (PGK). They identified a coevolving pair of positions which were important for the function of the protein, yet were not well-conserved, accepting aliphatic, aromatic or charged amino acids. They characterized two potential evolutionary pathways connecting different accepted pairs via both the covarion [11] and second site suppression [28] models of evolutionary change. Finally, a phylogenetic

analysis of coevolving positions in PGK revealed that putatively-coevolving positions are not characterized by either strong homoplasy or clade conservation (ie. synapomorphy and symplesiomorphy).

Also interestingly, they identified a region of high local covariation in the PGK alignment which could not be fixed via realignment [13]. Investigation of structure information in this region revealed that there were two alternative structural motifs in that region: some positions were α-helical and some were β-strands. While the sequence alignment program attempted to infer sequential homology in this region, the structures revealed that such alignment was not meaningful from a structural perspective. This observation reinforces the link between coevolutionary methods and the structural homology.

7.4 Software development

This PhD work represents a contribution of several bioinformatics software tools to the research community for rapid analysis of sequence data.

7.4.1 The MIpToolset for rapid calculation of covariation statistics

The MIpToolset [4] discussed in chapter 3 provides rapid calculation of the Mutual Information-derived covariation statistics described herein. It uses data structure optimizations to the Mutual Information algorithm to resolve the intrinsic memory and time challenges associated with such calculations. The result is a program that produces coevolution statistics, and adjoining figures and plot-ready data, with the efficiency required for database-wide analyses in reasonable time.

7.4.2 LoCo for alignment curation based on both conservation and local covariation

The algorithmic optimizations to the MIpToolset were used to facilitate the creation of LoCo [6], an alignment editor tool that is ideal for sequence alignment curation. LoCo is built on top of the extremely popular Jalview software [27] that has been cited over 2000 times, ensuring that the interface is instantly familiar to many. LoCo identifies putative systematic misalignments using local covariation. If the alignment does not show signs of increased local covariation, the researcher gains confidence in the conclusions drawn from the protein family. If the alignment has regions of high local covariation, a simple procedure exists for editing the alignment to correct the error. The speed of the MIpToolset allows covariation statistics to be updated in real time on screen below the alignment for typical protein families.

It is my hope that LoCo obtains widespread adoption as an intermediate validation after aligning sequences of a protein family, but before any conclusions are drawn from the alignment. The idea that an alignment should be viewed by the researcher before any subsequent analysis, is absolutely textbook. LoCo is the ideal tool for visualizing alignments at this stage because it increases the speed and accuracy with which a researcher can either approve or correct a new alignment. Correct alignments are crucial for making correct inferences about a protein family, and this work has shown that alignment errors can have serious unforeseen consequences to downstream analyses.

7.5 Future work

7.5.1 Improving multiple sequence alignment benchmarks

The field of sequence alignment is approaching the limits of some of the benchmarks available, like the aforementioned BAliBASE. Chapter 6 outlines an analysis of the BAliBASE dataset, for which numerous potential misalignments exist. Optimizing sequence alignment methods

against erroneous data will stifle innovation and advancement. Though labour-intensive, it is possible that a local covariation-validated database of alignments could be generated. This BAliBASE-LoCo database would potentially be more accurate and would open the door for new innovations in sequence alignment. Furthermore, it may be possible to automate sequence repair using local covariation; however, I hesitate to suggest this avenue because I believe strongly that everyone must visually inspect their data before moving on to later steps in the analysis pipeline. Automated alignment repair would actually save very little time in the typical case, and would encourage bypassing this crucial inspection step. Nevertheless, I acknowledge the utility of automated alignment curation for database-wide analyses.

7.5.2 Towards a universal benchmark for coevolutionary inference

Similarly, the lack of a universally-accepted benchmark database for coevolution is preventing optimal advancement of the field. However, the problem of creating an alignment-coevolution database is more complicated. The current benchmark method for inferring coevolution, contact prediction, is very coarse; it is not expected that all contacting positions will coevolve, nor is it predicted that all coevolving positions will be in contact. The expectation is only that coevolution often implies contact. However, the benefits of such a theoretical benchmark dataset would be enormous.

I believe that the path to such a dataset exists through investigating the contributions of putatively coevolving residues to the folding energy and relative motions within the protein structure *in silico*. The link between coevolving positions and thermodynamic coupling has been established [12]. I believe that is is possible to develop a metric which corresponds more strongly with coevolution than inter-residue contact. Such a metric would allow the separation of strongly interacting residues from those merely in contact and would allow for a more rigorous benchmark of coevolution statistics. It may also account for the observation that some covarying residues do so via indirect coevolution through an intermediate position [2]. Furthermore, such work could potentially lead to the precise mechanism of protein coevolution.

Such a mechanism would represent the answer to the near-half-century old question of the hypothesis which drives protein evolution: second site suppression [29] or covarions [11].

7.5.3 Creating one consensus statistic from many Mutual Information-based methods

As mentioned above, this work involves the development of two (Zpx, ΔZp) and further study of another (Zp) coevolution statistics; the consensus set of pairs between these methods represents the set of predicted contacting pairs with the highest accuracy. The consensus proves to be more accurate because each statistic selects a different, though overlapping, set of covarying pairs[7]. At present, there is not an established method for obtaining the optimal consensus between these three methods.

There are many possible avenues for developing such a methodology, including the development of a linear combination through some optimization algorithm. This optimization problem is, of course, tied to the previous goal of identifying a more accurate method for differentiating between true- and false-positives when inferring coevolving positions. The creation of an improved coevolution metric from Zp, Zpx, and ΔZp would benefit greatly from an established coevolution benchmark database. It would also be interesting to apply these statistics to the Critical Assessment of protein Structure Prediction (CASP) competition; CASP is a competition that allows structure prediction methods to generate structural predictions in advance of the publication of a previously-unknown structure [22].

7.5.4 Towards the detection of paralogous contamination

Finally, Chapter 6 highlights a region of high local covariation found in an alignment in the BAliBASE database that was associated with a paralogous alignment rather than with a putative misalignment. Mutual Information-based coevolution inference methods were designed with the assumption that the protein family being analyzed is orthologous [14]. After the gene

duplication event that creates paralogous sequences, each can undergo functional divergence which can lead to either a change in the evolutionary rate or a different complement of residues at that position [15, 16]. It is hypothesized that the functional divergence could cause a similar pattern of local covariation in the region of divergence[6], and thus has the potential to be used to identify putative binding or catalytic sites, when paralogous sequences are included on purpose, or the presence of paralogous sequence alignment contamination.

7.6 Final conclusions

In conclusion, I believe that the concept of coevolution, and the non-independence of positions within a protein family, is just as important to the understanding of a protein family as is the concept of conservation. The comparative popularity of conservation can be simply attributed to the relative simplicity of conservation as a concept. In order to improve the collective understanding of protein coevolution, and increase our ability to generalize it across major databases of protein families, I demonstrated the implicit link between the quality of coevolution predictions and the fundamental biochemical assumptions implicit in such analyses. I investigated the relationship between multiple sequence alignment methodologies and their effect on coevolution inference. This work led to an innovation in sequence alignment methods in the form of local covariation [7], and a user-friendly sofware tool, LoCo [6], which provides an interface for curating protein family alignments based on local covariation. With this new ability to generate high-quality sequence alignments for input into the coevolution inference analysis pipeline, new statistics with greater coevolution prediction accuracy and greater resistance to alignment errors were produced. These new innovations have been applied to the LAGLIDADG Homing Endonuclease protein family, and *in vivo* and *in vitro* experimentation supports the hypothesis that a coevolving network of residues constrains the variation of the family's active site. The advances in sequence alignment and coevolutionary inference presented here have the potential to provide new exciting insights into the evolution, structure,

and function of a diverse assortment of protein families.

Bibliography

[1] W R Atchley, K R Wollenberg, W M Fitch, W Terhalle, and A W Dress. Correlations among amino acid sites in bHLH protein domains: an information theoretic analysis. *Mol Biol Evol*, 17(1):164–178, January 2000.

[2] Lukas Burger and Erik van Nimwegen. Disentangling direct from indirect co-evolution of residues in protein alignments. *PLoS Computational Biology*, 6(1):e1000633, January 2010.

[3] J Z Dalgaard, A J Klar, M J Moser, W R Holley, A Chatterjee, and I S Mian. Statistical modeling and analysis of the LAGLIDADG family of site-specific endonucleases and identification of an intein that encodes a site-specific endonuclease of the HNH family. *Nucleic Acids Research*, 25(22):4626–4638, November 1997.

[4] R J Dickson and G B Gloor. The MIp Toolset: an efficient algorithm for calculating Mutual Information in protein alignments. *arXiv.org*, 2013.

[5] RJ Dickson, LM Wahl, AD Fernandes, and GB Gloor. Identifying and seeing beyond multiple sequence alignment errors using intra-molecular protein covariation. *PLoS ONE*, 5(6):e11082, 2010.

[6] Russell J Dickson and Gregory B Gloor. Protein sequence alignment analysis by local covariation: coevolution statistics detect benchmark alignment errors. *PLoS ONE*, 7(6):e37645, 2012.

[7] Russell J Dickson, Lindi M Wahl, Andrew D Fernandes, and Gregory B Gloor. Identifying and seeing beyond multiple sequence alignment errors using intra-molecular protein covariation. *PLoS ONE*, 5(6):e11082, 2010.

[8] S D Dunn, L M Wahl, and G B Gloor. Mutual information without the influence of phylogeny or entropy dramatically improves residue contact prediction. *Bioinformatics*, 24(3):333–340, January 2008.

[9] RC Edgar. Quality measures for protein alignment benchmarks. *Nucleic Acids Research*, 38(7):2145, 2010.

[10] Robert C Edgar. MUSCLE: a multiple sequence alignment method with reduced time and space complexity. *BMC Bioinformatics*, 5:113, August 2004.

[11] W M Fitch and E Markowitz. An improved method for determining codon variability in a gene and its application to the rate of fixation of mutations in evolution. *Biochem Genet*, 4(5):579–593, 1970.

[12] Anthony A Fodor and Richard W Aldrich. On evolutionary conservation of thermodynamic coupling in proteins. 279(18):19046–19050, 2004.

[13] GB Gloor, G Tyagi, DM Abrassart, AJ Kingston, AD Fernandes, SD Dunn, and CJ Brandl. Functionally Compensating Coevolving Positions Are Neither Homoplasic Nor Conserved in Clades. *Molecular Biology and Evolution*, 27(5):1181, 2010.

[14] Gregory B Gloor, Louise C Martin, Lindi M Wahl, and Stanley D Dunn. Mutual information in protein multiple sequence alignments reveals two classes of coevolving positions. *Biochemistry*, 44(19):7156–7165, May 2005.

[15] Xun Gu. Functional divergence in protein (family) sequence evolution. *Genetica*, 118(2-3):133–141, July 2003.

[16] Xun Gu. A simple statistical method for estimating type-II (cluster-specific) functional divergence of protein sequences. *Mol Biol Evol*, 23(10):1937–1945, October 2006.

[17] Kazuharu Misawa Kei-ichi Kuma Takashi Miyata Kazutaka Katoh. MAFFT: a novel method for rapid multiple sequence alignment based on fast Fourier transform. *Nucleic Acids Research*, 30(14):3059, July 2002.

[18] Ying Liu and Ivet Bahar. Sequence Evolution Correlates with Structural Dynamics. *Molecular Biology and Evolution*, April 2012.

[19] P Lucas, C Otis, JP Mercier, M Turmel, and C Lemieux. Rapid evolution of the DNA-binding site in LAGLIDADG homing endonucleases. *Nucleic Acids Research*, 29(4):960, 2001.

[20] Thomas A McMurrough, Russell J Dickson, Stephanie M F Thibert, Gregory B Gloor, and David R Edgell. A co-evolutionary barrier constrains active site variation in LAGLIDADG homing endonucleases .

[21] Faruck Morcos, Andrea Pagnani, Bryan Lunt, Arianna Bertolino, Debora S Marks, Chris Sander, Riccardo Zecchina, José N Onuchic, Terence Hwa, and Martin Weigt. Direct-coupling analysis of residue coevolution captures native contacts across many protein families. *Proceedings of the National Academy of Sciences*, 108(49):E1293–301, December 2011.

[22] John Moult, Krzysztof Fidelis, Andriy Kryshtafovych, and Anna Tramontano. Critical assessment of methods of protein structure prediction (CASP)—round IX. *Proteins: Structure, Function, and Bioinformatics*, 79(S10):1–5, 2011.

[23] Marco Punta, Penny C Coggill, Ruth Y Eberhardt, Jaina Mistry, John Tate, Chris Boursnell, Ningze Pang, Kristoffer Forslund, Goran Ceric, Jody Clements, Andreas Heger, Liisa Holm, Erik L L Sonnhammer, Sean R Eddy, Alex Bateman, and Robert D

Finn. The Pfam protein families database. *Nucleic Acids Res*, 40(Database issue):D290–301, January 2012.

[24] Ryo Takeuchi, Abigail R Lambert, Amanda Nga-Sze Mak, Kyle Jacoby, Russell J Dickson, Gregory B Gloor, Andrew M Scharenberg, David R Edgell, and Barry L Stoddard. Tapping natural reservoirs of homing endonucleases for targeted gene modification. *Proceedings of the National Academy of Sciences*, 108(32):13077–13082, August 2011.

[25] G Talavera and J Castresana. Improvement of phylogenies after removing divergent and ambiguously aligned blocks from protein sequence alignments. *Systematic biology*, 56(4):564, 2007.

[26] JD Thompson, F Plewniak, and O Poch. BAliBASE: a benchmark alignment database for the evaluation of multiple alignment programs. *Bioinformatics*, 15(1):87, 1999.

[27] Andrew M Waterhouse, James B Procter, David M A Martin, Michèle Clamp, and Geoffrey J Barton. Jalview Version 2–a multiple sequence alignment editor and analysis workbench. *Bioinformatics*, 25(9):1189–1191, May 2009.

[28] C Yanofsky, V Horn, and D Thorpe. PROTEIN STRUCTURE RELATIONSHIPS REVEALED BY MUTATIONAL ANALYSIS. *Science*, 146(3651):1593–1594, December 1964.

[29] C Yanofsky, V Horn, and D Thorpe. Protein structure relationships revealed by mutational analysis. *Science*, 146:1593–1594, 1964.

Appendix A

Reprint permissions

A.1 Chapter 2

http://www.elsevier.com/journals/journal-of-molecular-biology/0022-2836/guide-for-authors

Authors can use their articles for a wide range of scholarly, non-commercial purposes as outlined below. These rights apply for all Elsevier authors who publish their article as either a subscription article or an open access article.

We require that all Elsevier authors always include a full acknowledgement and, if appropriate, a link to the final published version hosted on Science Direct.

For open access articles these rights are separate from how readers can reuse your article as defined by the author's choice of Creative Commons user license options.

Authors can use either their accepted author manuscript or final published article for:

Use at a conference, meeting or for teaching purposes

Internal training by their company

Sharing individual articles with colleagues for their research use* (also known as 'scholarly sharing')

Use in a subsequent compilation of the author's works

Inclusion in a thesis or dissertation

Reuse of portions or extracts from the article in other works

Preparation of derivative works (other than for commercial purposes)

A.2 Chapter 3

http://arxiv.org/help/license

arXiv License Information

arXiv is a repository for scholarly material, and perpetual access is necessary to maintain the scholarly record. As such, arXiv keeps a permanent record of every submission and replacement announced.

arXiv does not ask that copyright be transferred. However, we require sufficient rights to allow us to distribute submitted articles in perpetuity. In order to submit an article to arXiv, the submitter must either:

grant arXiv.org a non-exclusive and irrevocable license to distribute the article, and certify that they have the right to grant this license, certify that the work is available under either the Creative Commons Attribution license, or the Creative Commons Attribution-Noncommercial-ShareAlike license, and that they have the right to grant this license, or certify that the work is in the public domain (we will store this information by associating the Create Commons Public Domain Declaration with the submission) In the most common case authors have the right to grant these licenses because they hold copyright in their own work. We currently support only two of the Creative Commons licenses. If you wish to use another license then it is appropriate to indicate a more restrictive version for arXiv records (both of the licenses we support give us sufficient rights to distribute articles) and then indicate the more permissive license in the actual article.

Note that if you intend to submit, or have submitted, your article to a journal then you should verify that the license you intend to select does not conflict with the journal license or copyright transfer agreement. Many journal agreements permit submission to arXiv with the

non-exclusive license to distribute which arXiv has used since 2004. The Creative Commons Attribution license in particular, permits commercial reuse and thus conflicts with many journal agreements.

A.3 Chapter 4

http://www.oxfordjournals.org/access_purchase/publication_rights.html

Rights retained by ALL Oxford Journal Authors

The right, after publication by Oxford Journals, to use all or part of the Article and abstract, for their own personal use, including their own classroom teaching purposes;

The right, after publication by Oxford Journals, to use all or part of the Article and abstract, in the preparation of derivative works, extension of the article into book-length or in other works, provided that a full acknowledgement is made to the original publication in the journal;

The right to include the article in full or in part in a thesis or dissertation, provided that this not published commercially;

A.4 Chapters 5 and 6

http://www.plosone.org/static/license

PLOS applies the Creative Commons Attribution License (CCAL) to all works we publish (read the human-readable summary or the full license legal code). Under the CCAL, authors retain ownership of the copyright for their article, but authors allow anyone to download, reuse, reprint, modify, distribute, and/or copy articles in PLOS journals, so long as the original authors and source are cited. No permission is required from the authors or the publishers.

In most cases, appropriate attribution can be provided by simply citing the original article (e.g., Kaltenbach LS et al. (2007) Huntingtin Interacting Proteins Are Genetic Modifiers of Neurodegeneration. PLOS Genet 3(5): e82. doi:10.1371/journal.pgen.0030082). If the item you plan to reuse is not part of a published article (e.g., a featured issue image), then please

indicate the originator of the work, and the volume, issue, and date of the journal in which the item appeared. For any reuse or redistribution of a work, you must also make clear the license terms under which the work was published.

This broad license was developed to facilitate open access to, and free use of, original works of all types. Applying this standard license to your own work will ensure your right to make your work freely and openly available. Learn more about open access. For queries about the license, please contact us.

Curriculum Vitae

Name:	Russell Dickson
Post-Secondary Education and Degrees:	University of Western Ontario London Ontario 2003-2008 B. Sc. University of Western Ontario London, ON 2008 - 2012 Ph.D.
Honours and Awards:	NSERC PGS M 2008-2010 NSERC PGS D 2010 - 2012 Graduate Student Teaching Award University Student Council Teaching Honour Roll
Related Work Experience:	Teaching Assistant The University of Western Ontario 2011 - 2012

Publications:

RJ Dickson, GB Gloor. (2013). Bioinformatics Identification of Coevolving Residues. Methods in Molecular Biology. In press.

RJ Dickson, GB Gloor. (2013). Gambling on Gaps: An Explanation of the Alignment-Coevolution Relationship. Bioinformatics. Under review.

RJ Dickson, GB Gloor. (2013). XORRO: Rapid Paired-End Read Overlapper. arXiv:1304.4620

RJ Dickson, GB Gloor. (2013). The MIp Toolset: an efficient algorithm for calculating Mutual Information in protein alignments. arXiv:1304.4573

RJ Dickson, GB Gloor. (2012). Protein Sequence Alignment Analysis by Local Covariation. PLoS One. 7(6): e37645.

R Takeuchi, AR Lambert, ANS Mak, K Jacoby, RJ Dickson, GB Gloor, AM Scharenberg, D.R Edgell, and B.L Stoddard. (2011). Tapping natural reservoirs of homing endonucleases for targeted gene modification. Proceedings of the National Academy of Sciences. 108(32): 13077

RJ Dickson, LM Wahl, AD Fernandes, and GB Gloor. (2010). Identifying and Seeing beyond Multiple Sequence Alignment Errors Using Intra-Molecular Protein Covariation. PLoS ONE.5(6): e11082

R Hummelen, AD Fernandes, JM Macklaim, RJ Dickson, J Changalucha, G.B Gloor, and G Reid. (2010). Deep sequencing of the vaginal microbiota of women with HIV. PLoS ONE. 5(8): e12078

GB Gloor, R Hummelen, JM Macklaim, RJ Dickson, AD Fernandes, R MacPhee, and G Reid. (2010). Microbiome profiling by Illumina sequencing of combinatorial sequence-tagged PCR products. PLoS ONE. 5(10): e15406

PW Pan, RJ Dickson, HL Gordon, SM Rothstein, S Tanaka. (2005). Functionally relevant protein motions: Extracting basin-specific collective coordinates from molecular dynamics trajectories. The Journal of chemical physics. 122: 034904