Summer 8-15-2017

Robust Algorithms for Detecting Hidden Structure in Biological Data

Roman Sloutsky
Washington University in St. Louis

WASHINGTON UNIVERSITY IN ST. LOUIS

Division of Biological and Biomedical Sciences

Computational & Systems Biology

Dissertation Examination Committee:

Kristen M. Naegle, Chair

Barak A. Cohen

Justin C. Fay

James J. Havranek

Gary D. Stormo

Robust Algorithms for Detecting Hidden Structure in Biological Data

by

Roman Sloutsky

A dissertation presented to
The Graduate School
of Washington University in
partial fulfillment of the
requirements for the degree
of Doctor of Philosophy

August 2017
St. Louis, Missouri

Table of Contents

Page

Page

List of Figures

Acknowledgments

First and foremost, I would like to thank my doctoral advisor and mentor, Kristen Naegle. She gave me tremendous freedom to explore and pursue my scientific interests, while providing invaluable advice and unwavering support whenever I needed it. This took a great deal of courage and trust, for which I am deeply grateful. The incredible enthusiasm she brings to all of her scientific endeavors is inspiring. Working with, and learning from Kristen has made me a much better scientist. In addition, she has been wonderfully open and honest about every aspect of being an independent investigator and running a lab, giving me unique insight, which I greatly appreciate.

I would like to thank the members of my thesis committee: Barak Cohen, Gary Stormo, Jim Havranek, and Justin Fay, for their guidance and insight, as well as for their patience. Individually, I thank Jim Havranek for inviting me into the Rosetta community and generously providing advice and support as I attempted to scale the infamous Rosetta learning curve. I thank Justin Fay for lending his insight and expertise any time I had questions about phylogenetic reconstruction. I thank Barak Cohen, both as chair of my thesis committee and as director of my graduate program, for his help and support in navigating a very difficult transition during my time in the program.

I would like to thank S. Joshua Swamidass for his advice on a variety of scientific and technical topics, as well as for the many frank, yet supportive conversations we had about

navigating a career in science. I have found Josh's unique perspective on the interface between academia and industry, not to mention life in general, to be very valuable.

I would especially like to thank my labmates Tom Ronan and Kathy Schaberg. Tom has been a wonderful colleague during my time in the Naegle lab and a great friend. In addition to our scientific discussions informing all of my work, Tom's rigorous and disciplined approach to scientific communication has helped improve all of my writing and presentations. Equally importantly, Tom's company has made my time in the Naegle lab tremendously enjoyable. Kathy has been generous and patient as she helped me become re-acquainted with the wet lab on each of my brief annual excursions, as well a kind and supportive friend.

I would also like to thank other current and former members of the Naegle lab. Alex Holehouse and Matt Matlock contributed scientific insight and technical advice as the programming aspects of my project became more challenging. I learned a great deal about software development from both. Ramya Palaniappan's infectious enthusiasm continues to remind me why I enjoy science. Mentoring Ramya, Elizabeth Worley, and Varun Krishnamurthy was immensely rewarding, and each contributed to various aspects of my project. In addition, I would like to thank former lab mates Zeke Maier, Drew Michael, Brian Haynes, and Molly Gibson for their friendship and support.

I would like to thank friends and colleagues Carlo Lapid, Ben Borgo, Adam Joyce and Darya Urusova, each of whom generously spent their time helping me with various aspects of my project. Carlo provided wonderfully detailed advice on statistical phylogenetics, both for my qualifying exam research proposal and, later, for my thesis project. Ben spent many

hours advising and encouraging me as I stumbled through Rosetta code and advised me in many aspects of setting up a peptide binding assay. Adam and Darya lent their considerable expertise as I learned to express and purify recombinant proteins. I would also like to thank Francesco Vallania for his help and advice in preparing for my qualifying exam and his help in preparing this dissertation.

I would like to thank Melanie Relich and Jeanne Silvestrini, both of whom served as my graduate program coordinators during my time with DBBS. They saved my proverbial bacon on more occasions than I can count or care to recall.

I have made wonderful friends during my time at Washington University. I particularly wish to thank the other two members of my program cohort: GiNell Elliott and Claire Schulkey, with whom I spent countless hours studying, writing, debugging C code, commiserating, and having fun. They have provided both scientific insight and emotional support throughout my PhD.

Finally, I want to thank my wife and family for their unending love and support throughout my time in graduate school. Hilary is a loving and endlessly patient wife, friend, and partner to me and a wonderful mother to our daughter Lily and our son Sebastian. This work was only possible because she cheerfully shoulders an unequal burden in making our family function. I can never thank her enough. My mom Olga, my dad Vladimir, and my sister Lena have given me unconditional love and a tremendous amount of help that I could not properly convey here. My parents-in-law Susan and John and my brothers-in-law Nicholas

and Christopher have been incredibly kind and supportive for as long as I have known them. I am immensely grateful to all of them.

Roman Sloutsky

Washington University in St. Louis

August 2017

Dedication

To Hilary, Lily, and Sebastian

I love you to the moon and back

Abstract

Robust Algorithms for Detecting Hidden Structure in Biological Data

by

Sloutsky, Roman

Doctor of Philosophy in Computational & Systems Biology,

Washington University in St. Louis, 2017.

Professor Kristen M. Naegle, Chairperson

Biological data, such as molecular abundance measurements and protein sequences, harbor complex hidden structure that reflects its underlying biological mechanisms. For example, high-throughput abundance measurements provide a snapshot the global state of a living cell, while homologous protein sequences encode the residue-level logic of the proteins' function and provide a snapshot of the evolutionary trajectory of the protein family. In this work I describe algorithmic approaches and analysis software I developed for uncovering hidden structure in both kinds of data.

Clustering is an unsurpervised machine learning technique commonly used to map the structure of data collected in high-throughput experiments, such as quantification of gene expression by DNA microarrays or short-read sequencing. Clustering algorithms always yield a partitioning of the data, but relying on a single partitioning solution can lead to spurious conclusions. In particular, noise in the data can cause objects to fall into the same

cluster by chance rather than due to meaningful association. In the first part of this thesis I demonstrate approaches to clustering data robustly in the presence of noise and apply robust clustering to analyze the transcriptional response to injury in a neuron cell.

In the second part of this thesis I describe identifying hidden specificity determining residues (SDPs) from alignments of protein sequences descended through gene duplication from a common ancestor (paralogs) and apply the approach to identify numerous putative SDPs in bacterial transcription factors in the LacI family. Finally, I describe and demonstrate a new algorithm for reconstructing the history of duplications by which paralogs descended from their common ancestor. This algorithm addresses the complexity of such reconstruction due to indeterminate or erroneous homology assignments made by sequence alignment algorithms and to the vast prevalence of divergence through speciation over divergence through gene duplication in protein evolution.

1. Introduction

By the standards of "big data"[1] even the most high-throughput biological datasets are trivially small. Yet such data can contain rich hidden structure reflecting the rich structure of the biological mechanisms that produced it. Extracting that structure can deliver profound biological insight. Mechanisms which produced the data, experiments by which the data was collected, and noise, both random and systematic, all contribute layers of complexity biologists have learned to appreciate. Much less appreciated is the fact that these layers of complexity can make our analysis algorithms behave in unexpected ways. This is particularly dangerous with algorithms which never fail to produce an answer, because it can be difficult to determine whether that answer is informative or misleading.

An organizing principle connects all of the work described in this thesis: understanding when the solutions our algorithms produce may be misleading and addressing that failure to get a more informative answer. In chapters 2 and 3 the analyzed data is numerical – quantified abundance of mRNA and protein post-translational modifications – and the algorithm is clustering. In chapters 4 and 5 the data is protein sequence and the algorithms are sequence alignment, phylogeny reconstruction, and detection of conservation patterns indicative of residue functionality. I developed methods for quantifying the reliability of

solutions produced by those algorithms and applying the same underlying algorithms to obtain more reliable and informative solutions.

1.1 Obtaining biological insight from the global state of a cell

Technological advances of the past 20 years allow collecting global snapshots of the state of a cell. Nucleic acids-based technologies – first DNA microarrays, then next generation sequencing (NGS) – have enabled profiling of global transcriptional[2,3], DNA methylation[4], chromatin accessibility[5], and protein-DNA interaction states[6]. Proteins[7], protein post-translational modifications[8], and metabolites[9] can be profiled by mass spectrometry. These data cannot be interpreted by eye. Obtaining biological insight requires computational analysis.

Clustering, an unsupervised machine learning technique, can identify patterns in complex data[10–12]. It has been successfully applied to learn genetic network architecture[13] and classify cancer types[14] from gene expression data, and to discover novel protein-protein interactions in signaling from phospho-proteomic data[15,16]. Gratifyingly, clustering algorithms always produce a partitioning of the data. However, the result may well be uninformative or even misleading[17].

One potential complication in clustering biological data is its high dimensionality when genes, proteins, or metabolites are treated as features on which the clustering is performed. Familiar distance metrics do not work as expected in high-dimensional spaces[18], causing many common clustering algorithms to produce meaningless partitions. Data objects may

have multiple, conflicting relationships in subsets of dimensions that are difficult to isolate without projecting the data into the right subspace[19]. Even in as few as two dimensions clusters may have complex shapes, rendering geometric centers meaningless and defeating centroid-based clustering algorithms, such as k-means[20].

Another complication is that high-throughput biological data are often noisy, which introduces uncertainty into the relationships between objects which clustering seeks to identify. Platform-specific noise models have been developed for DNA microarrays[21–23] and NGS[3], allowing proper treatment of noise prior to clustering, but to my knowledge such models do not exist for proteomic or metabolomic data collected by mass spectroscopy. Experimental schemes addressing specific sources of noise in NGS experiments, such as transcriptional biases [24] and sequencing errors[25] have also been developed. An ensemble clustering approach has been used to identify robust co-clustering relationships in proteomic data[15]. Fuzzy clustering has been used with metabolomic data[26], attempting to capture the underlying uncertainty of the data in the clustering solution itself. Nevertheless, experimental noise of various origins continues to complicate the interpretation of clustering results from biological data.

1.2 Learning about protein function from protein sequence

Protein sequences are data with incredibly complex structure. Since every aspect of a protein's function is encoded in its sequence, uncovering the hidden structure in this data is one of the grand challenges of modern biology. Here I discuss how evolution of protein

sequences impacts the cellular mechanisms that employ those proteins as components. I then discuss algorithmic approaches that have been developed to gain insight into how sequence encodes function and how sequence and function evolve together.

1.2.1 Gene duplication diversifies components of signaling and regulatory circuits

Nearly all of the work in cells is performed by proteins. Cells' regulatory and signal transduction mechanisms are no exception. Broadly speaking, regulatory and signaling proteins detect signals and perform functions in response to those signals. Signals may come in the form of post-translational modifications to one or more of the protein's residues, small molecule binding, or complex formation with one or more other proteins. The function performed in response may be catalysis, translocation to another cellular compartment, binding of DNA, simply providing a platform for other proteins to interact with each other, or any combination of those and other functions. In order to understand how the mechanisms function as wholes we must understand the function of their constituent parts, the proteins.

The number of tertiary folds available to proteins appears to be fairly limited[27,28], so evolution has adapted proteins that share a common fold to perform different functions. Frequently this happens in the context of a common global function performed with different specificities. Structurally similar enzymes catalyze the same reaction on different substrates[29,30]. Structurally similar transcription factors bind different DNA sequences and respond to different allosteric regulators[31]. In another strategy, metazoan proteins often

contain multiple independently folded domains, mixing and matching functions of individual domains for combinatorial effect. This is particularly true for signaling proteins which contain various combinations of catalytic, recognition, scaffolding, membrane-interacting, and other kinds of domains[32–34]. Some of the individual domains found in signaling proteins appear in over a hundred variations in a single genome[35,36], each acting on a different set of substrates, interacting with a different set of partners, or otherwise varying in their functional specificity. Since protein folds can tolerate a great number of amino acid sequences, these highly structurally similar proteins and protein domains can vary substantially in sequence. Nevertheless, their sequences are sufficiently similar to allow their identification by comparing their sequences to known representatives of the protein or domain family[37,38]. The origin of this shared similarity is common descent.

Theodosius Dobzhansky famously opined: "[n]othing in biology makes sense except in the light of evolution"[39]. Common descent is a fundamental model for understanding biology, and it applies not only to species and their genomes, but to individual fragments within those genomes as well. Genes encoding proteins with similar sequences and structures are believed to derive from a common ancestral sequence[40] (typically in the genome of an ancient ancestral species) which experienced a duplication event, giving rise to two identical copies. Given two identical genes able to perform the original's function, selection pressure to maintain that function is relaxed, allowing the two genes to sub-specialize or one to maintain the ancestral function while the other evolves a substantially different one[41,42].

Large families of protein domains evolved by numerous ancestral sequences undergoing this process multiple times, giving rise to domains with the finely-tuned functionalities. Furthermore, entirely new signaling systems likely evolved by this mechanism as well. In particular, phosphotyrosine-mediated signaling is thought to have emerged when the three components necessary to facilitate it – tyrosine kinases (writers), tyrosine phosphatases (erasers), and phosphotyrosine-recognizing SH2 domains (readers) – diverged from proteins with different functions: serine/threonine kinases, serine/threonine phosphatases, and a transcription elongation factor, respectively[36]. In fact, this development may have given rise to an entire new kingdom of life, Metazoa[35].

1.2.2 Sequence alignments have distinct evolutionary and structural interpretations

Sequence alignment, and multiple sequence alignment in particular, is an incredibly valuable tool for studying proteins. Because sequence alignment is broadly used by both evolutionary and structural biologists, both communities have shaped the development of alignment algorithms and software. However, the way sequence alignments are interpreted in the two fields are somewhat different.

The evolutionary interpretation of the mappings between individual positions in each sequence represented by their alignment is *homology* – descent, through mutation and selection, or lack thereof, from a specific position in their common ancestral sequence[43]. When a position in one sequence maps to a gap in another sequence, the evolutionary interpretation

is that an insertion or a deletion event occurred: either an insertion in the lineage leading to the sequence in which the position exists, or a deletion in the lineage leading to the sequence in which it does not.

The alternative interpretation of a mapping between sequence positions implied by alignment is structural correspondence, sometimes called structural homology[44], although the word "homology" itself is reserved by convention for the evolutionary interpretation[45]. Structural homology may or may not coincide with evolutionary homology. For example, independent insertion events may produce structurally homologous amino acid residues which do not share common descent and are, therefore, not homologous in the evolutionary sense. On the other hand, insertions elsewhere in one or both sequences can cause positions with a common ancestor to be structurally non-homologous. This is particularly common in disordered loops, which accommodate variable length better than secondary structure elements.

1.2.3 Multiple sequence alignment algorithms use heuristics to reduce complexity

Sequence alignment is a core tool in the study of molecular biology and evolution, and a over 100 methods for performing multiple sequence alignment (MSA) have been published to date[44]. These algorithms, their constituent components, and their performance have been reviewed and compared in great detail: see for example Edgar and Batzoglou[46], Notredame[47], Do and Katoh[48], Pei[49], Kemena and Notredame[50], Dessimoz and Gil[51], Thompson *et al.*[52], Löytynoja[43], Russel[53], and Chatzou *et al.*[44]. Briefly, MSA algorithms insert gaps into se-

quences to obtain an optimal set of position correspondences according to an objective function, which quantifies the quality of a proposed alignment in light of some model of the process by which the aligned sequences evolved. Most commonly the objective function is the sum-of-pairs over all pairwise mappings according to a substitution matrix (e.g. BLOSUM[54] or PAM[55]) for position-to-position matches and a gap scoring scheme, typically consisting of gap opening and gap extension components – so-called affine gap scoring[56].

Although a family of dynamic programing algorithms have long been known which guarantee the exact optimal pairwise alignment of two sequences for a given scoring scheme[43,56–58], their $O(l^n)$ complexity, where l is the average sequence length and n is the number of sequences, makes them computationally intractable as a general approach to multiple sequence alignment. In fact, finding the globally optimum alignment under sum-of-pairs objective functions is known to be an NP-complete problem[44,48]. Because of this, all MSA algorithms rely on heuristics, usually greedy heuristics[44,45,50], to search the space of possible alignments more efficiently, albeit without an optimality guarantee. Because of its generally superior performance on commonly used benchmarks[50], most modern algorithms incorporate the progressive alignment heuristic[59,60], which splits the overall alignment problem into a series of smaller alignment problems according to a guide tree, typically derived from exhaustive pairwise comparisons of the inputs[61]. Unfortunately, when progressive alignment algorithms incorporate errors at early steps, these errors propagate through the rest of the alignment process, which is informed by homology assignments made at previous steps. Iterative refinement schemes with alternating guide tree and alignment refinement and consistency-based

objective functions are sometimes employed, separately or in combination, to address this problem[43,48], though at additional computational cost.

1.2.4 Large Alignments Are Significantly Less Accurate

In light of the rapid increase in availability of biological sequences, two recent studies specifically addressed how alignment accuracy varies with the number of aligned sequences[62,63] and reached two main conclusions. First, almost all of the large panel of tested alignment tools, including all of the most accurate ones, failed to align the largest datasets (>10,000 sequences) and either failed or experienced impractically long running times (up to one month) when aligning between 5,000 and 10,000 sequences[62]. Second, alignment accuracy decreased with the number of aligned sequences. In a particularly elegant experiment, Sievers *et al.*[63] supplemented sequences from small curated structural alignments from alignment benchmarking databases BAliBASE3[64] and Homstrad[65] with increasing numbers of homologous sequences from Pfam[37], aligned the resulting set of sequences, and compared the accuracy of the embedded alignment of curated sequences to the accuracy of those sequences aligned alone by the same tool. All tested alignment tools experienced substantial, and progressively larger accuracy drop-offs with the addition of 500 or more Pfam sequences. A possible explanation for this is cumulative error propagation during progressive alignment increasing proportionately to the number of aligned sequences[50].

1.2.5 Phylogenetic reconstruction and sequence alignment inform each other

Phylogenetic inference always starts with a sequence alignment and is primarily concerned with the mutations which led to the observed differences between aligned sequences. The next step is typically the reconstruction of a phylogeny which describes the divergence of aligned sequences from their common ancestor. Most other phylogenetic analyses, such as detection of adaptive evolution[66–68], reconstruction of ancestral sequences[69], and inference of orthology and paralogy relationships between genes[42] require both an alignment and a phylogeny.

Inference of phylogenies for substantially diverged sequences, such as protein sequences in the "Twilight Zone" of homology, is extremely difficult[70]. While computationally efficient distance-based and parsimony-based methods perform with competitive or even higher accuracy for closely related sequences[71], the much more computationally expensive Maximum Likelihood and Bayesian statistical approaches outperform them on more distantly related datasets[72–74]. Statistical methods use explicit models of nucleotide, amino acid, or codon substitution and jointly infer the parameters of these models, the tree topology, and the branch lengths[75]. The need to use these methods makes accurate inference of large phylogenies of distantly related sequences particularly challenging[76].

Part the difficulty with inferring phylogenies of substantially diverged sequences is the frequent phylogenetic implausibility of alignments generated by general-purpose methods[77], particularly for large data sets (as discussed later in this chapter). This may result from the prevalent evaluation of alignment algorithms on structural alignment benchmarks, which

do not always accurately reflect evolutionary processes[78–80], for example by biasing representation to slowly evolving protein core regions at the expense of faster diverging coils[64], which are more likely to experience indel events[43,45]. Although general-purpose progressive alignment algorithms rely on guide trees, which are intended to account for evolutionary divergence[59,60], speed is generally prioritized over accuracy in guide tree inference. Generally some version of the distance-based Neibor Joining algorithm[81–83] is used, often resulting in inaccurate guide trees for substantially diverged sequences, which can in turn lead to alignment errors that cannot be fixed by iterative guide tree refinement[43].

Recognizing this weakness, a meta-method called SATé[84–86] was recently introduced, which co-estimates a phylogeny and an alignment by alternating between alignment with an accuracy-focused progressive alignment algorithm and phylogeny inference by Maximum Likelihood, using the inferred phylogeny as the guide tree for progressive alignment at the subsequent step. Although this approach can produce more accurate alignments than any progressive alignment method alone[84,85], it can still fail to produce a phylogenetically plausible placement of gaps[43,45]. PRANK[45,77] was developed specifically to address this concern by modeling gap placement in a "phylogeny-aware" manner. It performs well for phylogenetic applications[51,87,88], but is particularly sensitive to the accuracy of its guide tree[89] because of its use of inferred ancestral sequences to represent sub-alignments during progressive alignment[45].

The gold standard in accuracy remains Bayesian joint inference of alignment and phylogeny[90–93], which integrates over all alignments and all phylogenies in its search. Unlike

the sum-of-pairs objective functions used in other approaches, in this context the optimality criterion is maximal likelihood of the data – the alignment – given some parameters of an explicit character substitution model[43]. Unfortunately, due to their extreme computational complexity these methods remain limited to very small data sets[43,76].

Still, some general-purpose aligners work better for phylogenetic applications than others. In particular MAFFT's[61,94] accuracy-focused protocol, L-INS-i is consistently one of the best performers for downstream statistical phylogenetics applications[51,62,74,87], which is why it was selected as the default alignment algorithm in SATé[84,85].

1.2.6 Drawing functional inferences from protein MSAs

Under certain assumptions about how protein sequences evolve, multiple sequence alignments can be used to make functional inferences about homologous protein sequences. In statistical phylogenetics formal statistical frameworks exist for testing hypotheses about functional divergence of paralogous genes[95] and about correlated substitutions at multiple positions[96–98], the latter framework, covarions, having been introduced over 40 years ago. However, these approaches have not been adopted broadly, possibly due to the complexity of performing and interpreting the required statistical tests. They have not, to my knowledge, been used by molecular biologists. I do not discuss them here. Instead I focus on more commonly used heuristic approaches which reason about evolution implicitly by analyzing patterns in MSA columns.

Broadly, the heuristic approaches can be split into two categories: "unsupervised" methods, which identify correlations between amino acid patterns in pairs of alignment columns without using any information about the aligned sequences[99–121], and "supervised" methods, which identify correlations between patterns in individual alignment columns and classes into which the aligned sequences are grouped[122–145].

Unsupervised approach: identification of co-evolving residues

The underlying assumption of this approach is that natural selection may jointly constrain a pair of positions in a protein sequence, while not constraining either position to a specific amino acid. These methods can use information theoretic[108,114,117], statistical[101], correlation mode analysis[116], and maximum entropy[115,119], approaches among others. Because they analyze character patterns, reasoning about substitutions indirectly, they are forced to treat the phylogenetic relationship between sequences as noise, with the most accurate methods addressing this noise explicitly[114–117,119].

Although, in principle, residues need not be physically proximal to be under a joint substitution constraint, in practice the vast majority of such residue pairs form physical contacts in the protein's 3D structure. So prevalent are such pairs in fact, that these relationships have been used in a manner similar to NMR-derived distance constraints to determine protein folds *de novo*[146–148] and to map paths of transmission of information through protein cores between distant locations on protein surfaces[149,150].

Supervised approach: identification of functionally important residues

The underlying assumption of the supervised approach is that functional requirements constrain certain positions in protein sequences to specific amino acids, and that, therefore, such functionally important positions can be identified based on high degree of conservation. When the function under consideration is shared by all analyzed proteins, the analysis simply seeks the most conserved residues[151–153]. The more interesting case, however, is when the analyzed proteins share a global function, with different subsets of sequences performing the function with different specificities. In this case the positions of interest are expected to be conserved within each set of sequences sharing specificity, but not globally. Such positions are often referred to as specificity determining positions (SDPs).[122,124,126,127,129,130,132–135,137,139–145]

1.2.7 Effects of uncertainty and error in multiple sequence alignments

Because the true alignment of a set of biological sequences cannot be known, a constructed alignment of such sequences necessarily comes with some uncertainty. For example, in all but the most trivial cases different MSA algorithms will produce differing alignments of the same set of input sequences. Furthermore, most alignment algorithms hide the uncertainty of their own procedures by arbitrarily selecting among equally good solutions according to hard-coded rules, e.g. always deferring gap opening as late as possible, opting for reproducibility at the expense of accuracy[45,154]. Since the number of nearly equally good solutions is often very large[155,156], the preference for reproducibility masks non-trivial amounts of uncertainty.

Artifacts, such as the fact that most aligners produce different alignments when the input sequences are reversed[157], can also result.

Alignment errors propagate to downstream snalyses

Despite well known concerns about alignment uncertainty, downstream applications which require an alignment as input treat that alignment as an observation, assuming its correctness[158]. This problem has been widely recognized in phylogenetics, where a number of studies have demonstrated the sensitivity of downstream phylogenetic applications to differences in alignment of the same input sequences[62,72,74,88,158–161]. In fact, phylogeny reconstruction appears to be more sensitive to the method used to produce the input alignment than to the method used for the reconstruction itself[159], at least for some types of tree topologies[73]. Alignment errors also produce false positives when detecting positive selection in genes[87,88]. In one of the most comprehensive studies, which used both sequences simulated under known phylogenies and natural sequences with available high-confidence structural alignments, Wang *et al.*[74] found that, above a certain threshold of alignment accuracy, that accuracy was only weakly correlated with the accuracy of resulting trees. All "reasonably" accurate alignments resulted in quite accurate trees and differences between the alignments did not substantially affect tree topologies. However, for less accurate alignments the accuracy of the alignment correlated strongly with the accuracy of the inferred tree, with less accurate alignments resulting in significantly less accurate trees. This result makes sense when you consider that all phylogenetic tree inference methods in some way integrate signal

over every column in the alignment. Given an alignment with overall sufficiently accurate "phylogenetic signal" – sufficient fraction of correctly aligned columns – errors in the remaining columns can be tolerated without affecting the accuracy of the inferred tree.

Alignment curation

A number of tools have been developed to address this problem by identifying and removing alignment columns likely to contain errors, and, therefore, likely to decrease the accuracy of downstream analyses. Since actual alignment errors are impossible to identify, these tools use properties such as degree of column conservation[162], stability of pairwise position matches across varied gap scoring parameters[163] or guide tree topologies[164], statistically significant differences from randomly generated sequences[165], and the posterior probability of pairwise position matches derived from pair-HMMs[166] as proxies for likelihood of containing errors. Such filtering approaches can indeed improve the accuracy of phylogeny inferences[162,166].

Unfortunately, unlike phylogeny inference, column-wise functional analyses of alignments lose effectiveness with every alignment column containing errors, as results for positions aligned in these columns are likely to be erroneous. Nor can column-wise analyses take advantage of column filtering without sacrificing their effectiveness.

1.3 Research motivation

The central motivation of all work described in this thesis is uncovering the deeply hidden structure in biological data, improving and extending existing analysis algorithms when

their existing formulations cannot accommodate that hidden structure. This required developing means of quantifying the accuracy and reliability of solutions produced by the existing formulations, as well as of synthesizing those solutions into deeper insights.

The first two chapters concern analysis of numerical data reflecting global state of a cell. In chapter 2 I use an ensemble of clustering solutions using different data transformations, distance metrics, and clustering algorithms to analyze transcriptional response in a neuron cell in response to injury. In chapter 3 I discuss approaches to robustly handling experimental noise when clustering numerical data through several re-sampling strategies.

The next two chapters discuss approaches to analyzing multiple sequence alignments. In chapter 4 I demonstrate that protein families violate assumptions made in detection of specificity determining positions and, after correcting this shortcoming, identify numerous new putative specificity determinants in the LacI family of transcriptional regulators. In chapter 5 I use a novel approach to decomposition and synthesis of tree topologies in an algorithm for reconstructing the ancient history of duplication events that give rise to paralogous proteins.

2. Injury-Induced HDAC5 Nuclear Export Is Essential for Axon Regeneration

This chapter is adapted from part of the following published manuscript:

> Cho, Y., Sloutsky, R., Naegle, K. M. & Cavalli, V. Injury-induced hdac5 nuclear export is essential for axon regeneration. *Cell* **155**, 894–908 (2013)

2.1 Abstract

Reactivation of a silent transcriptional program is a critical step in successful axon regeneration following injury. Yet how such a program is unlocked after injury remains largely unexplored. We found that axon injury in peripheral sensory neurons elicits a back-propagating calcium wave that invades the soma and causes nuclear export of HDAC5. Injury-induced HDAC5 nuclear export enhances histone acetylation to activate a pro-regenerative gene-expression program. HDAC5 nuclear export is required for axon regeneration, as expression of a nuclear-trapped HDAC5 mutant prevents axon regeneration. These studies suggest a role for HDAC5 as a transcriptional switch controlling axon regeneration.

2.2 Introduction

Injured peripheral neurons successfully activate intrinsic signaling pathways to enable axon regeneration[168]. Within hours of a peripheral nerve injury, damaged axon tips are transformed into new growth-cone-like structures, and the expression of regeneration-associated genes in the cell body enhances axon regenerative capacity. In contrast, neurons within the central nervous system (CNS) typically fail at these tasks, leading to permanent neurological impairments. Defining how these intrinsic regenerative pathways are initiated may thus suggest therapeutic approaches to improve neuronal recovery following axon injury.

Activation of a genetic regeneration program is an important determinant of successful axon regeneration[169,170]. During development, multiple transcriptional pathways regulate the genes that control axons? intrinsic growth capacity. Once axons have reached their targets,

however, these transcriptional pathways are turned off, and the growth program is shut down. Peripheral neurons are able to successfully reactivate this growth program by expressing regeneration-associated genes that allow for robust axonal regrowth[169,170], whereas CNS neurons are typically unable to do so. Activation of such a pro-regenerative program is illustrated by the conditioning injury paradigm, in which a sensory neuron exposed to a prior peripheral lesion exhibits a dramatic improvement in axon regeneration compared to that of a naive neuron[169,171,172].

Although many studies have identified injury signals and transcriptional signaling pathways activated by nerve injury, the epigenetic mechanisms that control the switch from silent to growth-competent state following injury remain largely unexplored. Here we reveal that axon injury elicits nuclear export of histone deacetylase 5 (HDAC5), leading to enhanced histone acetylation. Promoting HDAC5 nuclear export mimics the conditioning injury paradigm and accelerates axon regeneration, whereas expression of an HDAC5 mutant that is retained in the nucleus prevents axon regeneration. Our results suggest that injury-induced HDAC5 nuclear export underlies an epigenetic switch controlling regenerative competence in adult sensory neurons.

2.3 Results

2.3.1 Axon Injury Stimulates HDAC5 Nuclear Export

Calcium influx and PKCμ are known to promote nuclear export of the class II histone deacetylase HDAC5 in cardiomyocytes[173] and hippocampal neurons[174], and we have shown

that PKCμ phosphorylates HDAC5 locally in injured axons[175]. Cultured dorsal root ganglia (DRG) neurons expressing GFP-HDAC5 were axotomized, and fluorescence intensity in the nucleus visualized over time. Control uninjured DRG nuclei displayed a stable level of fluorescence intensity, whereas axotomy induced a dramatic decrease in GFP-HDAC5 intensity in the nucleus.

2.3.2 HDAC5 Nuclear Export Is Required for Axon Regeneration

Peripheral nerve injury activates a pro-regenerative gene expression program that is essential to promote axon regeneration[169]. If HDAC5 nuclear export is required to activate such a pro-regenerative gene expression program, then preventing HDAC5 nuclear export should limit axon regeneration. We tested this possibility by engineering an HDAC5 mutant that is trapped in the nucleus and unable to be exported to the cytoplasm of DRG neurons. Based on previous studies[173,176], we mutated serine residues 259, 280, and 498 to aspartic acids (GFP-HDAC5nuc). In contrast to GFP-HDAC5, which reaches injured axon tips, GFP-HDAC5nuc was trapped in DRG nuclei and failed to reach axons following axotomy. DRG expressing GFP-HDAC5nuc also displayed decreased levels of acetylated histone H3 (Ac-H3) compared to DRG expressing GFP as a control, indicating that mutation of these serine residues affects GFP-HDAC5nuc localization but not its ability to regulate H3 de-acetylation.

We then monitored axon regeneration in vitro in DRG neurons expressing GFP-HDAC5nuc. We visualized axon regrowth by live-cell fluorescence imaging after in vitro axotomy in DRG

expressing GFP only or GFP together with GFP-HDAC5nuc and measured the regenerative capacity of injured axons after axotomy, as previously described[175]. Axotomized control axons displayed robust regeneration, with a regeneration index of 70.1% ± 4.1%, whereas GFP- HDAC5nuc expression strongly suppressed axon regeneration to 36.1% ± 6.9%.

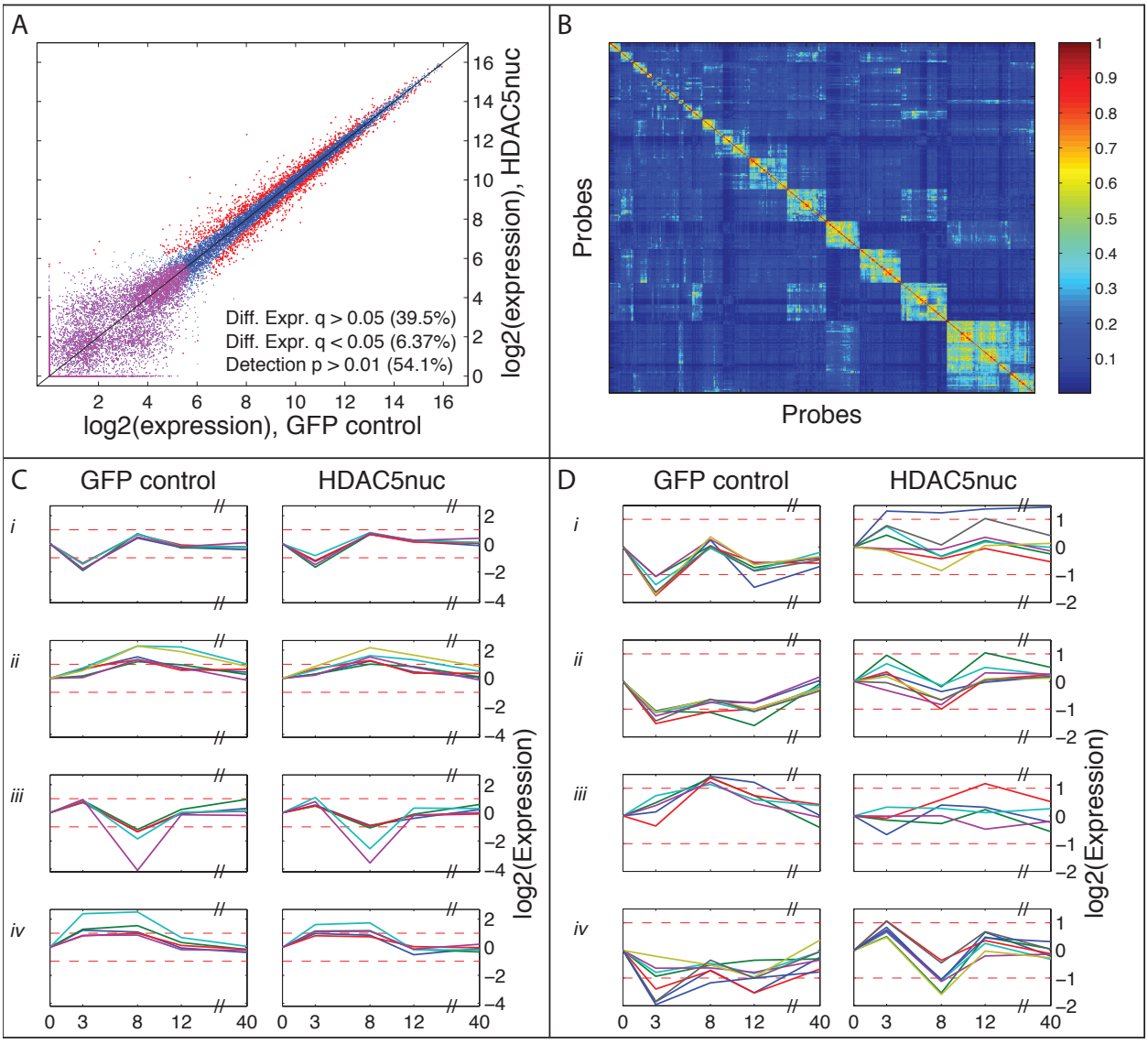

Figure 2.1.: Robust clustering of differentially expressed genes

Figure 2.1.:

Consistent with the idea that cytoplasmic localization of HDAC5 correlates with axon growth capacity, we observed that in freshly dissociated embryonic DRG neurons, which display high growth capacity, HDAC5 was mainly found in the cytoplasm, whereas, after 15 days in vitro, HDAC5 was mostly in the nucleus. Together these experiments point to a critical role of HDAC5 sub-cellular localization in the control of axon growth capacity. (**A**) Dependence of basal gene expression on GFP-HDAC5nuc expression. Red dots: differentially expressed probes with FDR-corrected q value <0.05 (1,637 probes, 6.4%); violet dots: probes below level of detection in both conditions (13,902, 54.1%); blue dots: remaining probes (10,158, 39.5%). (**B**) Heatmap representation of the pairwise co-clustering frequency matrix of 646 expression vectors corresponding to 323 probes in GFP- and GFP-HDAC5nuc-expressing DRG neurons. Ordering of probes along horizontal and vertical axis based on hierarchical clustering. (**C** and **D**) Post-axotomy time course dynamics of HDAC5-independent and HDAC5-dependent genes. Four example clusters each of HDAC5-independent (**C**) and -dependent (**D**) genes are shown. (**C**) Probes whose expression vectors in control and HDAC5nuc conditions co-clustered most frequently, suggesting that their post-axotomy dynamics are not regulated by HDAC5. Clusters i and ii exhibit rapid responses with a compensatory returns which first overshoot, then stabilize around basal expression level. Clusters iii and iv exhibit slower responses with less pronounced overshoot upon return to basal expression. (**D**) Probes whose expression vectors in control and HDAC5nuc conditions co-clustered least frequently, suggesting that their post-axotomy dynamics are subject to HDAC5 regulation. Clusters i and ii exhibit little response to axotomy under normal conditions, but a robust down-regulation in excess nuclear HDAC5. Clusters iii and iv exhibit the opposite: a robust post-axotomy down-regulation in control condition, but discordant mis-regulation in excess nuclear HDAC5.

2.3.3 HDAC5 Nuclear Export Activates a Pro-regenerative Transcriptional Program

To further determine the function of HDAC5 nuclear export in the expression of pro-regenerative genes following axon injury, we examined changes in gene expression in cultured DRG by microarray analysis, comparing DRG expressing GFP or GFP-HDAC5nuc at 0, 3, 8, 12, and 40 hr after in vitro axotomy. Comparison of pre-axotomy time points using differential fold-change analyses revealed that global expression differences were similar to those seen between replicates, indicating that gene expression was not generally affected by the expression of GFP-HDAC5nuc (Figure 2.1A). Several transcription factors, previously iden-

tified to play important roles in neuronal injury response and axon growth, were identified as HDAC5-dependent genes. These include *jun*[177,178], KLF4 and KLF5[179,180], *Fos*[181,182], and *Gadd45a*[183], and their expression was modulated by GFP-HDAC5nuc at one or more time points examined. This analysis suggests that injury-induced HDAC5 nuclear export plays an important role in the regulation of regeneration-associated genes.

To uncover dynamic temporal patterns of genes regulated by HDAC5, we conducted a robust clustering analysis on the 323 genes that exhibited a strong response to injury in either GFP- or GFP-HDAC5nuc-expressing neurons. A robustness metric for the similarity of temporal profiles was calculated by counting the number of times a pair of gene expression profiles co-clustered across all clustering sets (Figure 2.1B). Because we treated the temporal vectors of the GFP and GFP-HDAC5nuc conditions separately, but in the same clustering analysis, we were able to explore specific dynamics of patterns that were similar in the two conditions, versus different according to their co-occurrence in clusters (Figure 2.1C,D and Figure 2.1A). The patterns of genes uncovered in response to injury had varying temporal responses, consistent with previous reports[184,185]. These analyses support the notion that injury-induced HDAC5 nuclear export regulates the expression of genes important for axon regeneration.

2.4 Discussion

Injured adult peripheral neurons successfully regain growth competence via changes in gene expression to promote successful regeneration[169,170]. Yet little is known about the

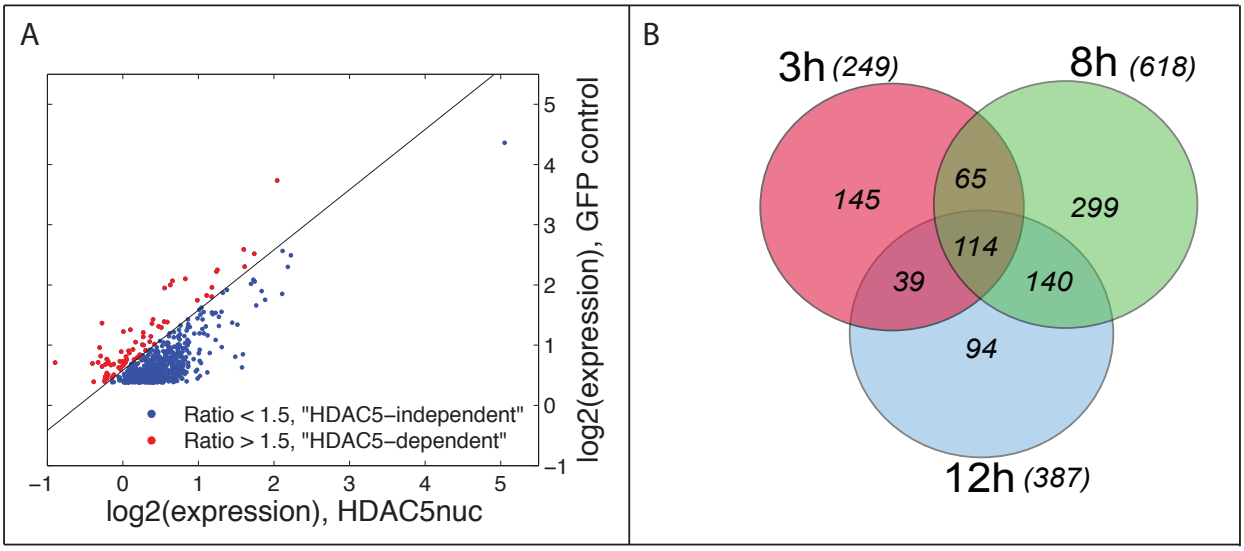

Figure 2.2.: HDAC5-dependent gene expression

(**A**) Plots of genes that respond significantly to injury at 8 hr. Red dots indicate those affected by GFP-HDAC5nuc expression, and the black line indicates a 1.5 ratio of GFP control neurons upregulation to GFP-HDAC5nuc neuron upregulation. (**B**) Venn diagram of HDAC5-dependent genes at the indicated time points after axotomy. The percentage of all genes at each time point that are HDAC5-dependent is indicated.

mechanisms by which injury signals unlock a silent pro-regenerative transcriptional program. Our study demonstrates that the changes are elicited via HDAC5 nuclear export.

2.4.1 Histone Modifications and the Tuning of Transcriptional Regenerative Pathways

The modification of histones by HATs and HDACs shapes chromatin to finely tune transcriptional profiles. Recent observations point to a role for histone modifications in the response of neurons to injury. Increasing histone acetylation promotes axon regeneration in CNS neurons, including cerebellar and retinal neurons[186,187]. In agreement with these studies, our results revealed that enhanced HDAC5 nuclear export in sensory neurons accelerates

axon regeneration. HDAC5 nuclear export may have a direct role in transcriptional regulation. Indeed, HDACs can deacetylate transcription factors in addition to histones and inhibit transcription via interaction with co-repressors[188]. Given that HDAC5 also functions as a repressor of the myocyte enhancer factor-2 (MEF2) transcription factor[189], injury-induced HDAC5 nuclear export may also regulate a pro-regenerative transcriptional program via transcriptional mechanisms.

HDAC5 nuclear export likely represents a part of an epigenetic response to injury. Indeed the role of DNA methylation has been suggested to regulate axon regeneration in the CNS[190]. Because chromatin remodeling plays an important role in neuronal function[188], future studies are needed to understand the epigenetic mechanisms induced by injury that promote axon regeneration in the adult nervous system.

2.4.2 Dual Role of HDAC5 in Axon Regeneration

We have previously shown that injury to peripheral neurons leads to HDAC5 accumulation at the tip of injured axons and local tubulin deacetylation, a process required for growth-cone dynamics and axon regeneration[175]. Here we present evidence that axon injury leads to export of HDAC5 from the nucleus to the cytoplasm. Our results strongly suggest that HDAC5 plays a dual role in peripheral axon regeneration: its exit from the nucleus permits activation of a pro-regenerative transcriptional program, and its transport in axons modulates growth-cone dynamics to sustain axon regeneration. This dual function of HDAC5 likely explains the decreased axon regeneration in HDAC5 KO compared to WT

mice. The complexity of roles of HDAC in neuronal development, function, and maintenance is rapidly coming to light, and future studies are needed to elucidate the multiple roles of distinct HDACs in axon growth and regeneration.

2.5 Experimental Procedures

2.5.1 DRG Culture, In Vitro Axotomy, and Regeneration Assays

Mouse embryonic DRG spot culture, in vitro axotomy, and regeneration assays were performed as described[175]. For in vitro regeneration assays, GFP-expressing DRG neurons were fixed at the indicated time after axotomy, and axons visualized by fluorescence microscopy.

2.5.2 Adult DRG Cultures and In Vivo Axon Regeneration Assay

For preconditioning injury, the sciatic nerves of 4-month-old mice were axotomized or not. L4 and L5 DRGs were dissected 3 days later, cultured for 8 hr, and immunostained with TUJ1, and axon projection length was calculated as previously described[191]. To test for axon regeneration in vivo, sciatic nerves were dissected 3 days after a crush injury. Longitudinal sections of fixed sciatic nerves were stained with SCG10 and TUJ1. SCG10 fluorescence intensity was measured along the length of the nerve using ImageJ and a regeneration index that was calculated by measuring the distance away from the crush site and in which the average SCG10 intensity is half that observed at the crush site.

2.5.3 RNA Preparation and Microarray

DRG spot cultures were axotomized at DIV7. RNA was extracted at 0, 3, 8, 12, and 40 hr after axotomy on duplicate samples using PureLink RNA extraction kit (Invitrogen). Quality control of extracted RNA was performed using 2100 Bioanalyzer (Agilent). RIN scores from all samples were more than 7.0 (minimum 8.8 and maximum 10). To analyze gene expression, MouseRef-8 v2.0 Bead-Chips were used from Genome Technology Access Center at Washington University. Data-quality assessment and normalization were performed using GenomeStudio (Illumina).

2.5.4 Data Analysis of Time-Course Dynamics

Background-subtracted data were normalized using the quantile algorithm[192] and in any given analysis if less than 40% of the measurements considered indicated values above the noise floor of detection (detection p-value > 0.01), the measurements were removed for the remainder of that analysis. Differential expression analysis and multiple hypothesis correction of significance values (q-values) using the Benjamini-Hochberg false discovery rate (FDR) were performed using Cyber-T software[193]. Probes whose basal expression was significantly affected by HDAC5nuc expression were removed from consideration (1,637 probes). For full-vector analyses, the remaining data set was reduced to include only probes whose expression changed by at least two-fold in response to axotomy in either the control or HDACnuc cells in the time points measured. The final reduced set included 323 probes. Raw expression values smaller than 1, including negative values, were set to 1 for the purposes of clustering analysis.

Replicate measurements, where available, were averaged. Control and HDAC5nuc condition vectors were treated independently during clustering (resulting in a matrix of 646 vectors across 5 time points). Multiple clustering analysis (MCA) was performed using MCAM[194] software in MATLAB (The MathWorks Inc., 2011). MCA was run with the following parameters: default (none), log2, Z score, normMax, pareto, normMax-log2, and Z score-log2 transformations; K-means, Affinity Propagation, Hierarchical, self-organizing maps (SOMs) and N-Cut clustering algorithms; Euclidean, correlation, city block, cosine, and Chebychev distance metrics; and K values ranging from 5 to 70 in increments of 2. The range of K values was centered around K = 37, which was the K value determined by running the Affinity Propagation clustering algorithm with cosine distance metric on log2-transformed data. The combined parameters produced a final MCA with 1,980 individual clustering solutions and the number of times any two probes co-clustered was summed across all clustering solutions to produce the co-occurrence matrix. To identify features of HDAC5-independent and -dependent transcriptional responses, probes with highly similar (co-clustering at least 75% of the time) and least similar (co-clustering no more than 15% of the time) dynamics in control and HDAC5nuc conditions were clustered again with the same clustering parameters but with ranges of K more suited to the sizes of these subsets, again determined by Affinity Propagation clustering with the same parameters. "Most similar" probes were clustered on data vectors consisting of all (averaged) measurements in both conditions, with K ranging from 3 to 15 in increments of 1, resulting in 780 clustering solutions. "Least similar" probes were clustered twice, once on control condition data and once on HDAC5nuc condition data,

with K ranging from 3 to 15 in increments of 1 in both cases, resulting in 780 clustering solutions. Probes failing to meet the two-fold expression change criterion in the clustered condition were removed prior to clustering, resulting in 62 probes in control condition and 27 probes in HDAC5nuc.

2.6 Acknowledgments

We thank Vitaly Klyachko for critical reading of the manuscript. We thank Eric Olson for the generous gift of the HDAC5 KO mice. We thank Domini Montgomery for technical support, Dennis Oakley for assistance with imaging, Ernie Gonzales and the Animal Surgery Core of the Hope Center for Neurological Disorders for assistance with optic nerve surgeries,We thank Dr. Seth Crosby and the Genome Technology Access Center at Washington University for microarray analysis. This work was supported in part by grants from NIH (DE022000 and NS082446) and the McDonnell Center for Cellular and Molecular Neurobiology.

3. Accounting for Noise When Clustering Biological Data

This chapter is adapted from the following published manuscript:

> Sloutsky, R., Jimenez, N., Swamidass, S. J. & Naegle, K. M. Accounting for noise when clustering biological data. *Brief Bioinform* **14**, 423–36 (2013)

3.1 Abstract

Clustering is a powerful and commonly used technique that organizes and elucidates the structure of biological data. Clustering data from gene expression, metabolomics, and proteomics experiments has proven to be useful at deriving a variety of insights, such as the shared regulation or function of biochemical components within networks. However, experimental measurements of biological processes are subject to substantial noise—stemming from both technical and biological variability—and most clustering algorithms are sensitive to this noise. In this paper we explore several methods of accounting for noise when analyzing biological datasets through clustering. Using a toy dataset and two different case-studies—gene expression and protein phosphorylation—we demonstrate the sensitivity of clustering algorithms to noise. Several methods of accounting for this noise can be used to establish when clustering results can be trusted. These methods span a range of assumptions about the statistical properties of the noise, and can therefore be applied to virtually any biological data source.

3.2 Introduction

High-throughput experimental technologies that capture large numbers of molecular measurements, such as gene expression, metabolomics, and proteomics technologies, are increasingly common in routine biological research. In order to understand the data in high-throughput biology, researchers often use clustering algorithms to organize, visualize and in-

fer relationships between objects (e.g. proteins, genes, or samples) within a high-dimensional dataset.

A variety of clustering algorithms have been employed to analyze biological data, such as Hierarchical clustering and K-means clustering. See Jain et al.[196] for a detailed review. Clustering algorithms partition a dataset into clusters where measurements within a cluster are more similar to each other than they are to members of other clusters. The similarity measure is based on a distance metric, such as the Euclidean distance. This improves our ability to visualize complex data by reducing the number of objects into a smaller number of clusters. Doing so helps us understand the underlying process which generated the data. Clustering has been used in a variety of contexts for elucidating a variety of biochemical processes[13,14,26,197].

However, biological data is noisy and clustering algorithms are sensitive to this noise. Noise in experiments arises from the techniques used to make those observations, human error and variability, and the intrinsic stochasticity of the system itself. Certainly, experimental design and technological advances can reduce biological noise, but there usually remains some non-negligible uncertainty about each measurement. The best way to quantify this uncertainty is to replicate the measurement several times[21,198]. Analysis done using these measurements should account for measurement uncertainty. Unfortunately, most clustering algorithms do not explicitly account for the underlying uncertainty of measurements.

Our goal is to explore how noise within real datasets impacts the clustering results and interpretation of clustering. We will cover four methods of accounting for noise, which

can be combined with any clustering algorithm of choice. These methods span a range of assumptions regarding the independence of measurements and requirements for the number of replicates. The focus of this work is on algorithm-independent methods that can easily be combined with virtually any commonly-used clustering algorithm. First, we present a toy example to demonstrate explicitly how noise affects a controlled clustering problem. Next, we will introduce four example methods of accounting for noise. Finally, we will discuss two case studies using real biological data: a phosphoproteomic dataset of insulin signaling and a dynamic microarray experiment of EGF-induced gene expression.

3.3 Toy Example

The toy example in Figure 3.1A could represent a variety of experiments, such as the measurement of 100 mRNA transcripts in cancer cells versus normal tissue, or of 100 metabolites in untreated versus cells treated with a drug. Using this toy model we can start to understand how uncertainty in experimental measurements affects our confidence in the clustering solution.

The toy system is made up of five Gaussian processes, with 20 points generated from each process. We will refer to this as the 'true' data. The clusters in Figure 3.1A were generated by K-means clustering with K=5. This solution matches the underlying processes. Black lines indicate cluster boundaries, such that for every point inside a cluster's boundary area its Euclidean distance to that cluster's centroid is smaller than its Euclidean distance to any other centroid. Of course, true empirical measurements would have some noise associated

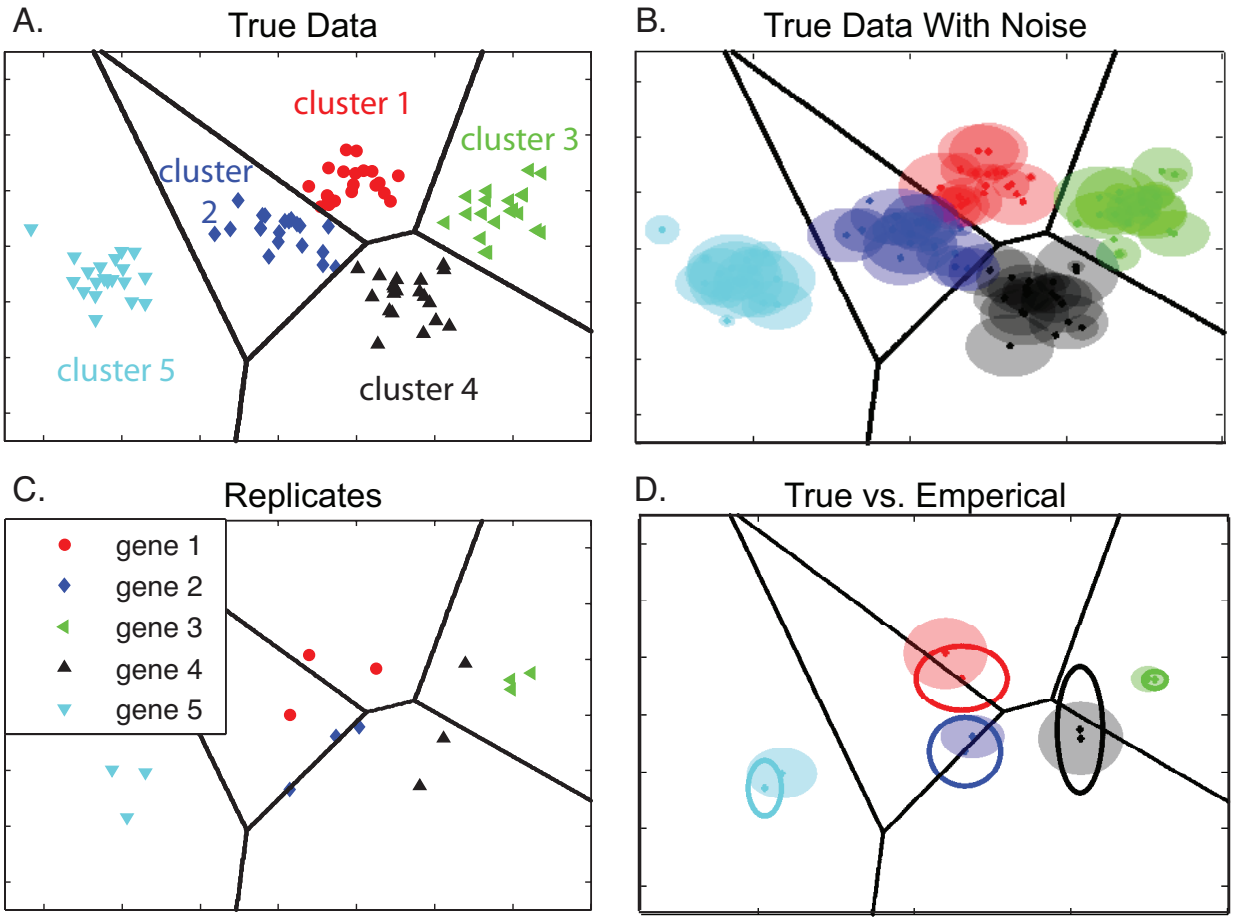

Figure 3.1.: Toy clustering example

(**A**) Five clusters were generated by sampling from five different 2-dimensional Gaussian distributions. The cluster labels and boundaries are generated from a K-means solution that recapitulates the true clusters. (**B**) Gaussian noise was randomly generated for every individual data point and is shown here by shading the area that captures one standard deviation in both dimensions. (**C**) A toy triplicate experiment was generated by randomly choosing two points in addition to the mean of the distributions shown in panel B. For simplicity, only one object is shown per cluster and those where triplicates cross clustering boundaries were chosen. For descriptive purposes, they are referred to here as "genes". (**D**) Empirical distributions are calculated from the individual triplicates and are shown outlined in their respective cluster color with the true distribution represented by shaded area.

with them, so noise has been superimposed on each of the 100 data points in the toy dataset. Shaded areas in Figure 3.1B indicate one standard deviation of each noise distribution. It now becomes obvious that noisy measurements near dense cluster boundaries, such as those between clusters 1, 2 and 4, could lead to misinterpretations of the relationships between members of these clusters.

The most common approach to clustering experimental data with replicates is to cluster means of replicate measurements. See Method A in the next section. Important information may be lost when replicates are condensed in this manner prior to clustering analysis. For example, outlier points may seriously diminish or overemphasize a relationship between objects. Some data points may simply mis-cluster because their measurement average does not accurately reflect the underlying data.

In order to test typical clustering approaches on our toy dataset, we generated two additional replicates for each point in the original dataset by randomly drawing from the noise distributions indicated in Figure 3.1B. For visualization purposes, replicates of only one object from each cluster are plotted in Figure 3.1C. While the objects from clusters 3 and 5 had no misclassified replicates, objects from clusters 1, 2, and 4 had at least one replicate which mis-clustered. Figure 3.1D shows the resulting empirical means and standard deviations (outlines) compared to the means and standard deviations of underlying noise distributions (shaded regions). This is a demonstration of how limited replicates may not accurately reflect the true underlying process. Although the average value representations of most of the objects in Figure 3.1D clustered accurately, the average of the object from cluster

2 is mis-clustered. Additional experimental replicates improves our confidence regarding measurement accuracy, and can also improve our confidence in clustering solutions. Please see Dougherty et al.[199] for an in-depth discussion of the relationship between experimental replication and clustering precision.

We repeated the *in silico* experiment of generating two additional replicates from the noise models in Figure 3.1B and clustering the empirical averages. On average we observed three objects per experiment, out of 100 total, mis-clustered when represented by empirical averages. Mis-clustering occurred for objects in all clusters except cluster 5. This demonstrates that impact of noise on clustering is influenced by the distribution of data in the multidimensional space. Measurements for objects in cluster 5 could effectively tolerate higher amounts of noise without impacting their association with that cluster, whereas measurements falling close to boundaries were often assigned incorrectly in the clustering solution. For real biological data, unlike this toy example, knowing whether correct partitioning has occurred is impossible. The probabilistic nature of the result from this type of noise analysis underscores the facts that hard cluster boundaries may not be meaningful and that measurements with noise, which span multiple cluster boundaries, could be considered to belong partially to multiple clusters.

3.4 Clustering Strategies

Some mixture-model based clustering methodologies have been developed which solve for clustering solutions while taking noise or replicates into account[200–202]. However, there

Example Data Matrix

$\boldsymbol{D}$ has size j x k for each replicate (n),
For example, expression for 100 genes measured across 7 time points in triplicate ($j = 100$, $k = 7$, $n = 3$)

Method A: Replicate averaging

Calculate the average value of each replicate, $\boldsymbol{D}^*$ with size j x k
$C =$ cluster($\boldsymbol{D}^*$)

Results for Toy Data: Assigned to the correct cluster?

Gene 1	Gene 2	Gene 3	Gene 4
Yes	No	Yes	Yes

Method B: Replicate co-clustering

$C =$cluster($\boldsymbol{D}^*$), $\boldsymbol{D}^*$ has size n*j x k
Evaluate C for replicate co-clustering stability

Results for Toy Data: Percentage of correctly clustered replicates

Gene 1	Gene 2	Gene 3	Gene 4	Gene5
67%	33%	100%	67%	100%

Method C: Permutation sampling

Repeat:
  Make $\boldsymbol{D}^*$ (with size j x k) sampled from replicates
  $C_i =$ cluster($\boldsymbol{D}^*$)
Evaluate the ensemble of cluster solutions

Results for Toy Data: Percentage of correctly clustered replicates

Gene 1	Gene 2	Gene 3	Gene 4	Gene5
56%	46%	100%	66%	100%

Method D: Model-based sampling

Establish a statistical distribution for measurements in **D**
Repeat:
  Make $\boldsymbol{D}^*$ (with size j x k) by sampling from distribution
  $C_i =$ cluster($\boldsymbol{D}^*$)
Until: Satisfy number of repetitions or a convergence criteria
Evaluate the ensemble of cluster solutions

Results for Toy Data: Percentage of correctly clustered replicates

Gene 1	Gene 2	Gene 3	Gene 4	Gene5
48%	44%	100%	58%	100%

Figure 3.2.: Robust clustering methods

We examine four methods of clustering. The results of five genes from the toy dataset (Figure 3.1C) are reported. In this case, we know the true correct cluster of each gene, and report the percentage of times that each Gene is correctly classified. In practice, the percentage of times each gene pair clusters together would be reported.

are clustering methodologies which may work particularly well for a given type of data, or which a researcher may be particularly well-equipped to implement, for which a model-based incorporation of noise or replicate handling does not exist. Therefore, we are going to focus on methods that can be used in combination with any clustering algorithm and chosen set of clustering parameters.

We consider four methods of handling noise in clustering (Figure 3.2). In Method A, the data is collapsed by averaging each replicate experiment, and this averaged data is clustered. In Method B, the complete data with all the replicates expanded is clustered, and the concordance of replicate clustering is quantified, see related work in Yeung et al.[203]. Methods C and D are different ways of "ensemble" clustering, which are combinations of clustering new instances of a data matrix, which themselves are likely representations of the

data. In Method C, new data matrices are formed by shuffling the data between replicates. In Method D, new data matrices are produced by sampling from an analytic distribution of the data. Examples of handling noise by ensemble clustering can be found in Kerr et al.[23] and Bittner et al.[204].

While most methods of accounting for noise in clustering can be viewed as special cases of these four methods, this section does not constitute a fully comprehensive review of the field. Rather, the focus of these methods is on testing a single clustering algorithm's sensitivity to noise. We do not address whether a particular clustering solution is in fact optimal for a given dataset or desired information outcome. Moreover, we do not discuss previously developed methods which directly modify clustering algorithms, in non-trivial ways, to handle noise. Some of these methods can be found in the following references: Medvedovic et al.[200], Ng et al.[201], and Cooke et al.[202]. In this way, we hope to focus on the methods most broadly applicable across a wide range of biological analysis.

3.4.1 Method A: Clustering Replicate Averages

The majority of studies cluster biological data one time on single vector representations of the data. In the case where multiple replicates exist, it is common to use the average of the replicates to represent the data. This method of average-value clustering will be referred to as Method A, Figure 3.2A. If there are enough replicates, this is a reasonable way of managing experimental noise, because the average of the replicates converges to the average of the true distribution. In high-throughput biology experiments, however, there are usually

a limited number of replicates. With few replicates, as we will see, this is a poor method of managing experimental noise.

3.4.2 Method B: Replicate Co-Clustering

The second method works by clustering all the data, with all the replicates, and measuring the robustness of a result by quantifying if replicates of each object are placed in the same cluster.

The advantage of this method is that it is easily implemented and evaluated. A visual demonstration of replicate co-clustering is shown in Figure 3.1C for a subset of the 100 objects. The results are also summarized in Figure 3.2B, which indicate the percentage of times the five example genes have replicates in the correct cluster. It immediately becomes clear that we have gained important information about the potential misclassification of Gene 2 that occurred from clustering the average of the replicates, which is indicated by only one of the three replicates correctly being assigned to cluster number 2. Additionally, Genes 1 and 4 each have one mis-clustered replicate, despite the average values being correctly classified.

The disadvantage of this method is that the degree of associations between measurements and clusters can take on only finite values, such as 0%, 33%, 67%, and 100% in the case of triplicates. This becomes even more problematic in the case of duplicates, where different clustering results for two replicates would not allow for the selection of a single cluster by majority vote. Regardless of its limitations, this is still an important improvement

over average-value clustering since it can highlight the potentially non-robust assignment of certain measurements. Other methods which directly use replicates have been used. For example in Yeung et al.[203], they explore forcing replicates into the same subtree as a seed for further Hierarchical clustering.

3.4.3 Ensemble Clustering

In ensemble clustering, the clustering algorithm is applied to resampled versions of the data to generate multiple clustering solutions. These clustering solutions are combined to create a consensus, or ensemble solution[20,205–208]. The critical reason ensemble clustering is more powerful than replicate co-clustering is that it enables generation of many more samples than there are replicates. This enables better resolution of co-clustering confidence. Furthermore, as we shall see, this method can be applied even to single-replicate datasets.

The ensemble has been used to address a variety of issues that arise in clustering including the effect of initialization on non-deterministic algorithms (such as K-means)[20,209,210], sensitivities to algorithm, distance metric, and data transformation selections[211] and incorporating the effect of noise[23,204,207,212]. In essence, the ensemble has been used to address how to handle variations in clustering results that arise from the factors that could alter the solution, such as the distance metric used or data variability.

Different groups have used different methods of generating an ensemble solution in the context of noise sensitivities[23,204]. For example, Bellec et al.[212] use ensemble clustering to find stable features of brain networks in resting-state fMRI with sampling from noise distributions

of data and allowing the initialization of K-means to change randomly. Unfortunately, studies like this, which account for the effect of experimental noise on a clustering solution, are not published frequently enough.

In this discussion we focus on a variation of the method used by Bellec et al.[212] because of its ease of implementation. In this method, clustering results are combined to create an ensemble solution by computing a co-occurrence matrix, which indicates the fraction of times each pair of objects cluster together across all clustering sets. Objects that co-cluster frequently are said to "robustly" cluster, and are the connections that can be most trusted.

Several groups are exploring methods of evaluating the co-occurrence matrix generated from the cluster ensemble. These methods range from linkage-based clustering of the co-occurrence matrix to define a final ensemble clustering solution[20,207,210,213,214] to graph theoretical-based methods, where the co-occurrence matrix is viewed as a weighted association matrix between objects[215].

An advantage of ensemble clustering is that the final ensemble result can take on shapes that are different from the constraints of the underlying clustering algorithm used[209]. For example, as we will see in the case studies, the network visualization of the co-occurrence matrix can have a different number of clusters than the K-means clustering algorithm form which it was sampled. Additionally, an important piece of information contained in the ensemble is not just the decrease in probability of one object clustering with a second object, but in balance, what other clusters that object could alternately be associated with.

Ensemble clustering naturally lends itself to a probabilistic, fuzzy clustering interpretation. When the same clusters are consistently identified across clustering sets, the probability of an object belonging to a cluster is simply estimated by the frequency of this occurring in the ensemble, thereby defining fuzzy cluster boundaries. For many real datasets, however, cluster identities cannot be mapped between clustering sets because the identified clusters are so different between clustering solutions. Nevertheless, co-clustering frequencies between pairs of objects may be treated as probabilities of belonging to the same cluster. Robust clusters may then be built up from the most robust pairwise relationships.

We will introduce two methods of producing an ensemble result that accounts for noise. In the first method, the replicates themselves will be reshuffled to produce a new dataset. In the second method, an alternate dataset will be created by sampling from a noise model. The case studies have been chosen to explore various nuances of these methods, including how to deal with single-replicate data.

3.4.4 Method C: Permutation Sampling

In permutation sampling, which could be considered a form of bootstrapping, the data vectors for each object's replicates are shuffled to generate each sample (Figure 3.2C). This is done by randomly picking a value of each measurement to form a novel replicate. Exhaustively enumerating all permutations of the dataset's replicates is computationally expensive for realistically-sized datasets.

A permutation sampling method is attractive because it is easy to implement, and makes no explicit assumptions regarding the underlying distribution of the data samples. It is not entirely assumption free. It implicitly assumes, for example, that sample to sample noise for a given object is independent, an assumption sometimes violated if data is not effectively normalized. Also this method assumes there are sufficient replicates to adequately sample each object. The problem with this requirement is subtle. There is a large number of objects in typical biological datasets, so with a handful replicates it is almost certain that a few of these objects will have non-representative samples[216].

3.4.5 Method D: Model-Based Sampling

In the final and most powerful method samples are drawn from a mathematical model of the data-generating process that explicitly defines the experimental noise. This model is computed from the observed data and appropriate, domain-specific assumptions about the noise.

Sampling consists of generating random values from probability distributions characterizing the replicate data for each point in each vector to create a resampled dataset, **D***. Normal distributions are used when normality conditions apply to the data. These distributions can be defined by the means and standard deviations of replicates, or may be inferred from the data by more sophisticated methods[216]. However, experimental data may deviate from normality and other parametric models of noise can be used for sampling. The advantage of this method is that if a noise model can be defined for a certain process, even datasets

with no replicates can be evaluated for the effect of noise on the analysis by clustering. The potential limitation of this application is when assumed distributions deviate drastically from the sampled distribution leading to a skewed, or over- or under-representation of the noise.

3.4.6 Modeling Noise

In order to use Method D, some effort must be made towards defining an appropriate model of the experimental uncertainty in the data. When there are two or more replicates, it is commonly assumed that the noise is normally distributed and that each object's noise is independent. The sample mean and standard deviation of each object are used to parameterize normal distributions from which sampling replicates are drawn[217] . For this to be sensible, the data must be transformed onto a scale where the data is approximately normally distributed. For example, expression data is usually normalized and transformed onto a log-scale to accomplish this.

With domain- and technology-specific studies, it is possible to use better models. In the case of gene expression, several researchers have proposed better noise models[22,216,218]. For example, Posekany et al.[22] argues that microarray noise has a fatter tail than a normal distribution, and suggest using a t-distribution instead.

The sliding-window prior, proposed by Baldi and Long[216], merits special attention because it is likely to be applicable across several types of biological data. They note that there is a strong, non-linear relationship between the mean and standard deviation of genes in expression data (Figure 3.3A). Furthermore, when there are few replicates, the most prob-

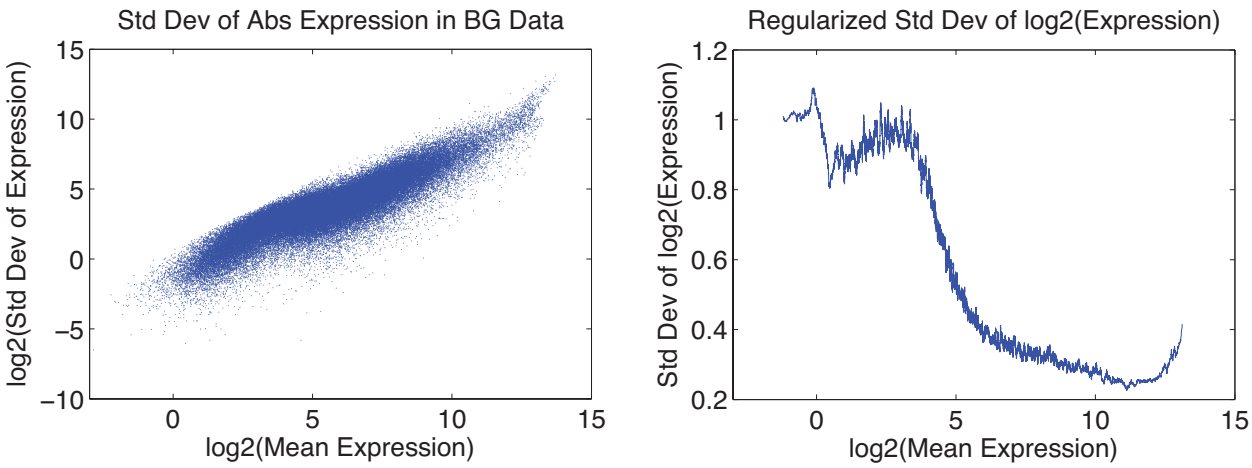

Figure 3.3.: The relationship between mean and variance

Left: In gene expression data (shown here), there is a strong relationship between the mean expression and the sample variance. The higher the expression level, the higher the variance. **Right**: In log-space, the relationship flips. The higher the expression level, the lower the variance. The sliding-window prior averages the variance of genes with similar expression levels. The figure displays variance estimates as a line.

lematic error in modeling is underestimating sample variance, which implies there is more confidence in the true mean than is warranted. In order to mitigate the risk of dramatically underestimating the variance, they propose averaging the variance of genes with similar mean expression levels (Figure 3.3B) . There is often a strong, but non-linear, dependence between the mean and variance in real biological data, so this regularization is often sensible.

Most methods of modeling noise require that there is at least one replicate of the experimental data. Replicate datasets are not always collected. It is still possible to model noise in this context if the technology is well understood. An example of this is demonstrated in the second case study.

3.5 Case Studies

To illustrate how these method work in practice, we will present two case studies, one using phosphoproteomic profiling data and one using gene expression data.

3.5.1 Case 1: Phosphoproteomic Data with Replicates

This case study focuses on a quantitative LC-MS/MS phosphoproteomic experiment, which captures phosphotyrosine signaling dynamics in 3T3-L1 adipocytes stimulated with insulin[219]. The dataset was downloaded from PTMScout[194] for analysis and it represents 120 phosphopeptides measured at 0, 5, 10 and 30 minutes after stimulation with insulin. Although biological triplicates were measured, due to technological limitations, 15% of the phosphopeptides have no replicate information and 29% of the phosphopeptides are only measured in duplicate. Since this dataset contains replicates, we applied all four methods to this dataset in order to compare the results of each, for a particular dataset and clustering implementation. In this section, we will present results relative to Method A results, i.e. clustering using the average of the replicates.

Method A: Clustering Replicate Averages

Hierarchical clustering, with a Euclidean distance metric and average linkage was chosen as the method for clustering. The dendrogram was cut such that 12 clusters were formed (i.e. K=12). This set of clustering parameters was chosen based on relatively good performance for producing clusters enriched for biological terms, such as Gene Ontology labels (deter-

mined by using the PTMScout interface[194]). The heat map of the co-occurrence matrix for average-value clustering is shown in Figure 3.4A, a matrix of ones and zeros. The size of each cluster is shown in a bubble diagram in Figure 3.5. The clustering set solution for this implementation is composed of two large clusters, four smaller clusters, and outlier clusters composed of, at most, two members. The two largest clusters can be seen in the upper-right and lower-left portions of the co-occurrence matrix, Figure 3.4A.

Method B: Replicate Co-Clustering

Since 85% of the dataset has at least one replicate, we applied Method B and clustered all replicates together to see how co-clustering was affected, Figure 3.4B. Using the same order of phosphopeptides as shown in Figure 3.4A, this heat map clearly shows the two largest clusters have replicates which cluster between both groups, which is indicated by the appearance of clustering between the upper-left and lower-right clusters, Figure 3.4B. Additionally, the next two largest clusters also have replicates co-clustering between the two groups. The two smallest clusters, which are separated by the clusters made of single members, do not appear to change as dramatically in structure when replicates are considered. For those phosphopeptides with at least one replicate, 27.5% of them have replicates which cluster differently. Even this simplistic attempt at considering noise within an experiment has informed our understanding about the relationship amongst clusters beyond that of average-value clustering. In particular, it indicates that replicates between the largest groups co-cluster when considered as individual vectors within clustering.

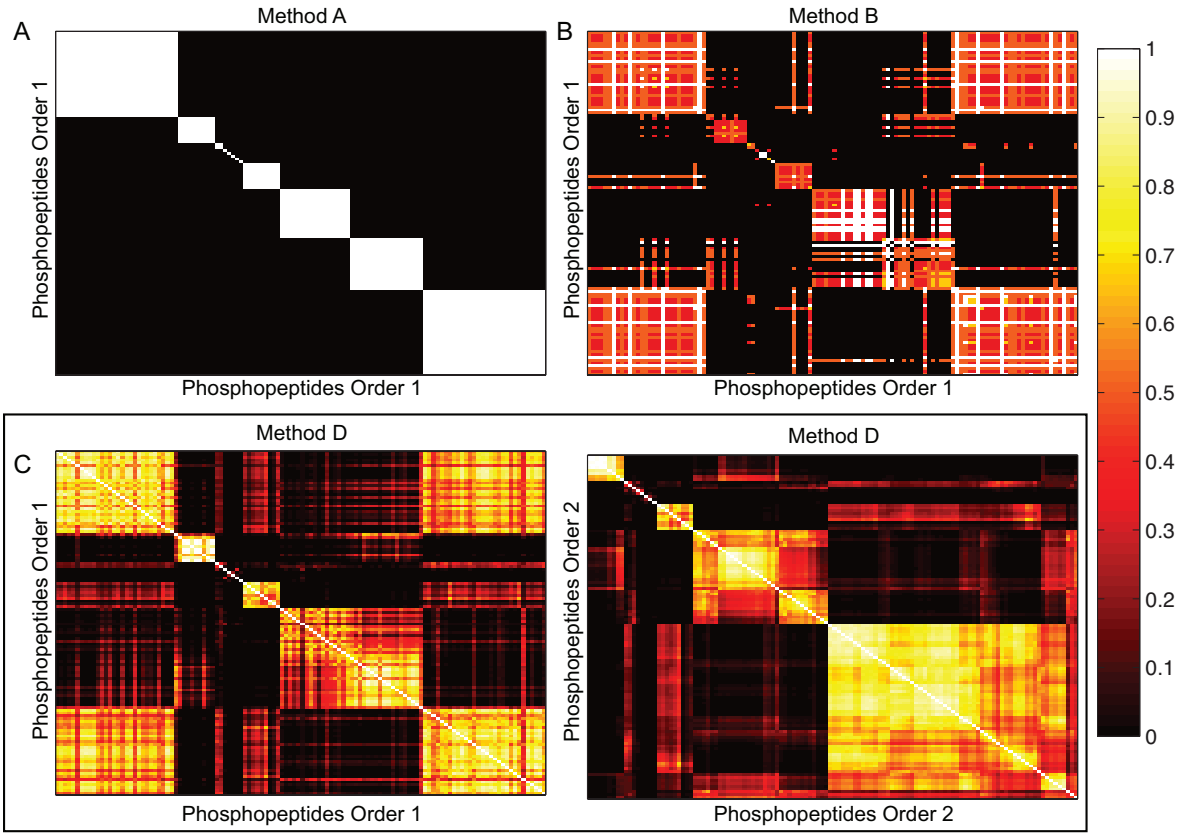

Figure 3.4.: Co-occurrence Matrices

(**A**) Co-occurrence matrix from clustering averaged data (Method A). A co-occurrence matrix was generated from the single clustering result of the averaged data matrix, using Hierarchical clustering, K=12, of 120 phosphopeptides measured across four time points. Co-occurrences amongst all members of a cluster is set to a value of 1 and was re-ordered according to linkage by the Ward algorithm, this order is referred to as Order 1. (**B**) Co-occurrence Matrix of co-clustered replicates (Method B). The number of replicates co-clustered between phosphopeptides was summed and normalized by the minimum number of replicates between the two phosphopeptides. This matrix is plotted in the same order as panel A. The same two large clusters appear in the upper left and lower right area of the heat map, but one can see that those clusters have members that have replicates co-clustered, as indicated by the lower left and upper right portions of the heat map. Since phosphopeptides have one, two or three replicates, co-occurrence values can only take on a value of 0, 1/3, 1/2, 2/3 or 1. (**C**) The co-occurrence matrix of an ensemble result (Method D). Co-occurrences are summed between pairs across all clustering results of an ensemble formed by sampling from a normal distribution defined by the replicate mean and standard deviation. The matrix is shown in two phosphopeptide orderings: the same order as panel A and one defined by a new linkage using the Ward algorithm.

Ensemble Results: Methods C and D

Ensemble clustering by permutation and model-based sampling were accumulated across 5000 iterations. Figure 3.4C shows the co-occurrence matrix that results from sampling a normal distribution model of noise, defined by a measurement's experimental mean and standard deviation. For measurements with no replicates the experimental mean was assumed to be the single replicate data and the standard deviation was derived from a global estimate of the ratio of sample standard deviation to sample mean, known as the coefficient of variation (CV). Global CV was estimated by averaging CV's of measurements with replicates. This simple approach assumes a linear relationship between variance and mean across the dataset, but a more complicated model could be fit to that relationship if necessary. Standard deviations for measurements without replicates were calculated from their means and the global CV estimate. The co-occurrence matrices were only subtly different between the two methods of sampling. The approximate behavior of the ensembles is the same as co-clustering replicate data; those cluster boundaries which break down when replicates are considered are also blurred when ensemble clustering is used. In this method probabilities of pairwise relationships can take on a finer range of values versus simply clustering replicate data directly.

As mentioned earlier, there are many methods for finding a single clustering solution from an ensemble. Here we used the Ward algorithm to hierarchically cluster the co-occurrence matrix and chose a cutoff to assemble eight clusters[20], which appeared to be the naturally occurring breakdown of the co-occurrence matrix for both Methods C and D (Figure 3.4).

Figure 3.5 illustrates the results of the sampling methods; both compared to the cluster structure of Method A results. The basic structures match well to the information contained in the heat maps of the co-occurrence matrices (Figure 3.4), i.e. the two largest clusters are remixed to varying extents in the ensemble result and there are two relatively stable smaller clusters (clusters 3 and 5 in the middle panel). It becomes clear from this illustration that one of the strengths of ensemble clustering is the ability to naturally capture outliers (cluster number 4). These phosphopeptides are outliers based on two pieces of evidence, they do not cluster robustly with any other phosphopeptide and they undergo drastically different dynamics compared to the rest of the dataset (Figure 3.5). Interestingly, despite being outliers, when joined by ensemble analysis, they form a potentially biologically meaningful subset of phosphorylation sites belonging to proteins involved in regulation of vesicle fusion.

Mapk1 T183/Y185 (Mapk1-p2), Mapk1 Y185 (Mapk1-p1) and Irs1 Y935 are particularly interesting examples to explore in more depth since their cluster membership changes with consideration of noise. Mapk1-p2 and Mapk1-p1 are members of a relatively stable cluster enriched for members of the MAPK cascade, however, permutation sampling indicates that the singly phosphorylated form of Mapk1 instead resides in the cluster containing Irs1 Y935. The disparate results amongst the methods can be more clearly understood by observing the data used in clustering for each method (Figure 3.5). The middle panels of Figure 3.5 demonstrate that Mapk1-p2 and Irs1 Y935 are relatively low variance measurements, with distinctly different down-regulation profiles. In contrast, Mapk1-p1 replicates have much higher variance, and that variance acts in such a way as to make its association with either

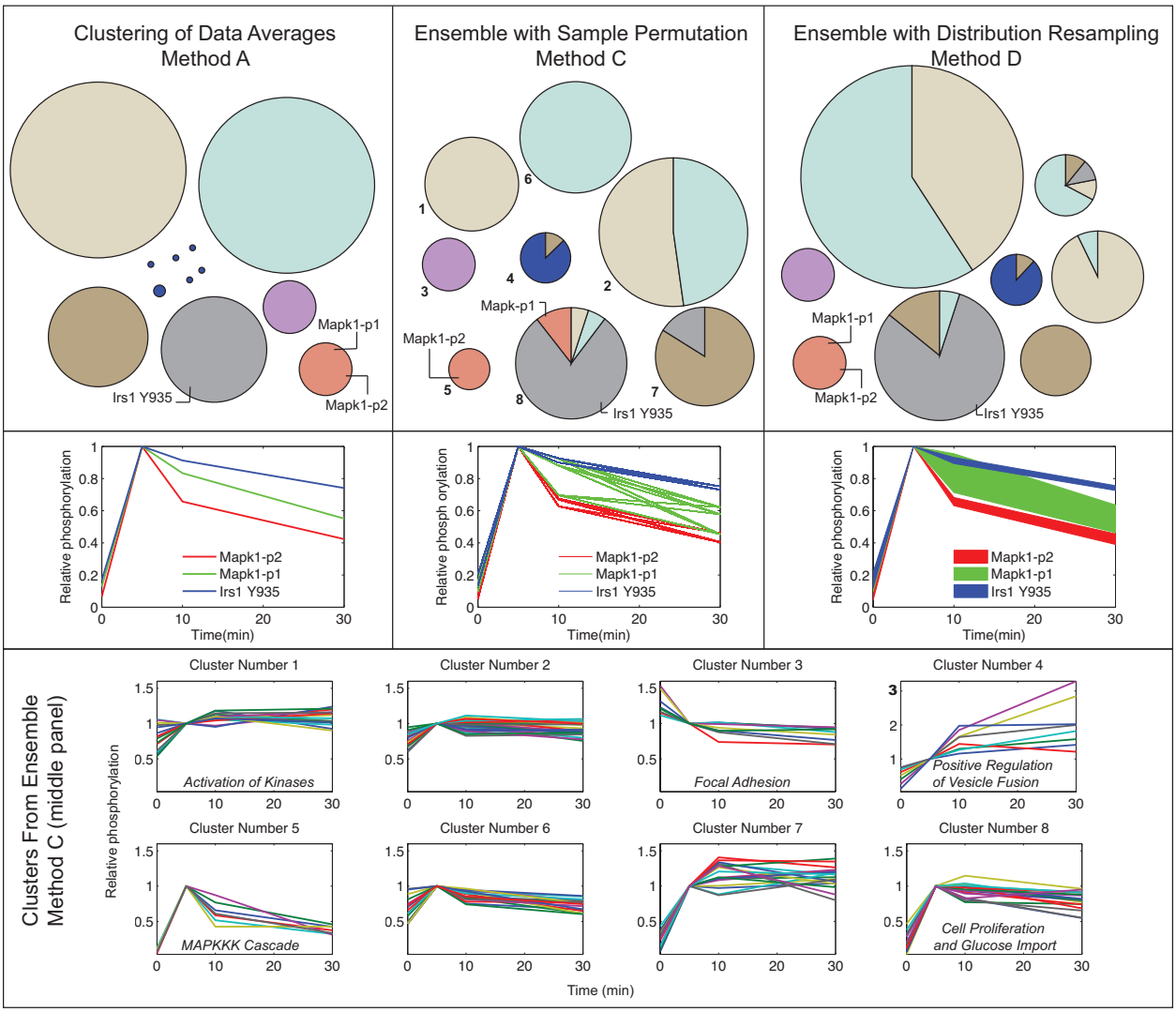

Figure 3.5.: Phosphoproteomic data clustering results

Upper panels: Ensembles formed from data permutation and normal distribution sampling compared to a single clustering of the averaged data matrix. Cluster sizes are proportional to the number of phosphopeptides contained; the smallest contains one phosphopeptide (left panel) and the largest contains 35 phosphopeptides (right panel). A single clustering was produced for the ensemble sets by cutting the clustered co-occurrence matrix (Figure 3.4D) to form eight clusters. **Middle Panels:** Dynamics of three example phosphopeptides, Mapk1 T183/Y185 (referred to as a doubly-phosphorylated form of Mapk1, Mapk1-p2) and Irs1 Y935 robustly cluster in their respective groups. The middle panel of the Mapk and Irs1 phosphopeptide dynamics show all possible permutations of their replicates. The right panel of the Mapk and Irs1 phosphopeptide dynamics indicate the area that contains +/- a standard deviation of all vectors produced by sampling 5000 times from the normal distribution defined by their respective replicates. **Lower Panel:** The average representation of the dynamics in each cluster produced by Method C, Upper Middle Panel (cluster numbers map directly between the Upper and Lower panels). If Gene Ontology enrichment occurs in the cluster, a representative label appears. Note that cluster 4 y-axis scale is double that of the other clusters.

Irs1 or Mapk1-p2 indeterminable. Both Mapk1 and Mapk3 singly and doubly phosphorylated forms behave identically, indicating that this could in fact represent biologically meaningful information. In cases such as these, it is perhaps best to view these as fuzzy clustering relationships. In this way one would describe the singly phosphorylated forms of Mapk1 and Mapk3 as belonging partially to the cluster containing Irs1 Y935 and partially as belonging to the doubly phosphorylated forms of the Map kinases.

Since this dataset represents a real-world example with an unknown "ideal" clustering solution, it is impossible to say which method of handling noise is best. However, from this study it is possible to see that average-value clustering is the least informative when it comes to understanding the robustness of a given solution with regards to experimental noise. The remaining three methods, clustering replicates and the ensemble methods, follow surprisingly similar trends when it comes to highlighting relationships that are not robust to noise. The advantage of the ensemble methods appears to be their ability to define a finer range of co-clustering values, which could be helpful in defining either an improved, definitive clustering solution, or a fuzzy clustering solution.

3.5.2 Case 2: Gene Expression Data Without Replicates

In this case we examine a single measurement microarray gene expression experiment. HeLa cells were stimulated with epidermal growth factor (EGF) for 0, 20, 40, 60, 120, 240, and 480 minutes, followed by gene expression profiling by hybridization to the Affymetrix

HG-U133A array[220]. The dataset is publicly available from www.ncbi.nlm.nih.gov/geo/, record GSE6783.

The subject of this study was transcriptional response, and the authors focused their analysis on putative regulators of transcription induced by EGF. To model the intent of the original study, we performed ensemble clustering on a subset of 655 probe sets meeting the following criteria: 1) both "DNA" and "transcription" appear in Gene Ontology annotations of the quantified transcripts, and 2) at least a two-fold increase in expression was observed at any time point between 20 and 240 minutes, relative to the basal condition.

In contrast with Case 1, no replicate data was collected, a common scenario in high-throughput biological experiments. The absence of replicate data restricts our methodological options.

Methods A, B, and C

In the absence of replicate measurements Methods A, B, and C cannot be used to account for noise. Unlike Case 1, where averaging replicate measurements constitutes an accounting of noise, albeit a naive one, clustering single measurements completely ignores it. In this case Method D, employing sampling from probability distributions, must be used to account for noise.

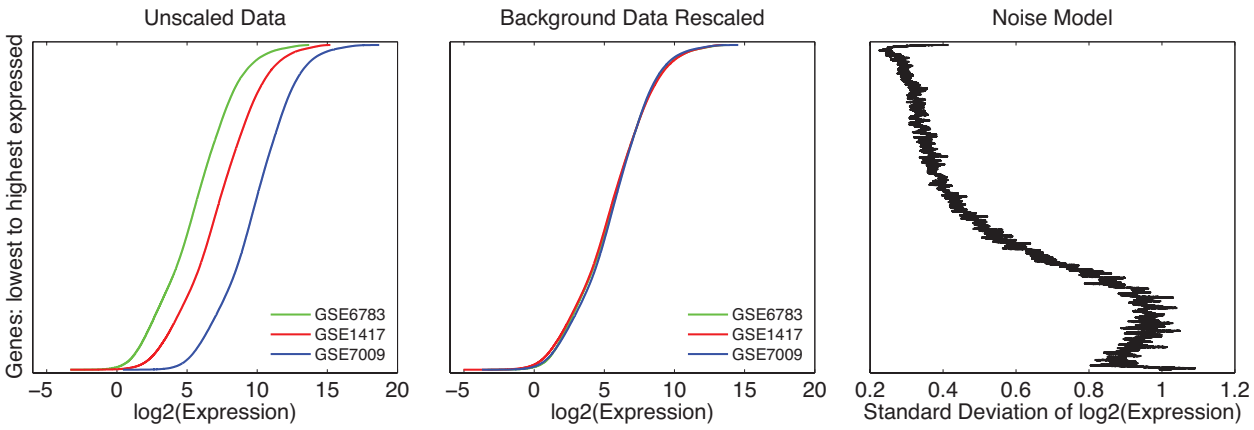

Figure 3.6.: Noise model preparation

Left panel: Unscaled foreground (GSE6783) and background (GSE1417 and GSE7009) data, ordered by expression level. Although internally consistent, expression levels cannot be compared between datasets. **Middle panel:** Background data rescaled to the 75th quantile of foreground data. Expression levels across all quantiles now comparable between datasets. **Right panel:** Regularized noise model derived from background data using Cyber-T. In log-space noise is higher for genes with low expression levels.

3.5.3 Sampling Without Replicate Data

Single measurements can be used to estimate positions (first moments, expectations, or means) of probability distributions from which the measurements were drawn. However, they provide no information about the shapes (second moments, or variances) of those distributions. In order to estimate the variances we used, as 'background', other data collected on the same microarray platform to create a mapping between means and variances, based on the assumption that there is a strong and non-linear relationship. Microarray experiments assay gene expression globally, but the majority of genes will not change significantly in any single experiment. Therefore, we assume that we can use datasets collected from the same cell line and on the same microarray platform to generate a model for noise in the absence of replicates in a particular experiment.

We selected two expression datasets, which were also collected in HeLa cells and on the Affymetrix HG-U133A platform, accessible from www.ncbi.nlm.nih.gov/geo/ as records GSE1417[221] and GSE7009[222]. The first experiment was measured in triplicate and the second in quadruplicate. Since measurements are not comparable between datasets without adequate scaling (Figure 3.6, left panel), background datasets were rescaled by the factor (75th quantile expression in foreground/75th quantile expression in background) to map them to the expression range of the 'foreground' (GSE6783) experiment, while preserving all pairwise fold-differences between probe sets within each background dataset. Rescaling makes all three datasets comparable across the entire range of expression (Figure 3.6, middle panel).

For the purpose of this demonstration we modeled the noise of log-transformed expression data with normal distributions[216,223]. From rescaled background data we generated a mapping from mean log2(expression) to standard deviation (Figure 3.6, right panel) using Cyber-T software[216], which employs the sliding window prior to calculate regularized standard deviation estimates for each set of replicates under normality assumptions. This mapping was interpolated to select standard deviation values for each foreground experimental measurement.

Normal distributions may not adequately model noise for some microarray expression data[224], and heavier-tailed t-distributions have been proposed as a suitable alternative[22,225]. Background data may be similarly used to parametrize a t-distribution noise model, for

example using the algorithm developed by Posekany et al.[22], which may then be interpolated to obtain distribution parameters for foreground data.

Method D

Replicate measurements were generated for the 655 selected probe sets by sampling from the normal distributions parametrized as described above. Those samplings were then clustered by K-means with cosine distance metric and K=20. Here, we present results from 500 iterations of sampling, followed by clustering, since co-clustering frequencies >0.5 did not change appreciably with larger number of replications. K=20 was selected because it roughly fits the square root of number of objects, a general rule of thumb for selection of K, and it seemed to produce relatively well-formed clusters.

Only 11% of probes met the threshold requirement of co-clustering with any other probe 50% of the time or more. A robustnesscutoff of >0.65 was chosen for determining robust clusters, since it maximized the number and size of distinct clusters, shown in Figure 3.7. More stringent choices for the threshold significantly dissipated the formation of robust clusters and lowering the stringency resulted in clusters that were too large to interpret. At this cutoff, Clusters 1 and 2 are still clearly distinct (Figure 3.7, bottom panel), although AKAP17A may be thought of as partially belonging to both. As discussed earlier, robustness analysis via noise sampling ensembles naturally produces fuzzy clusters with probabilistic boundaries.

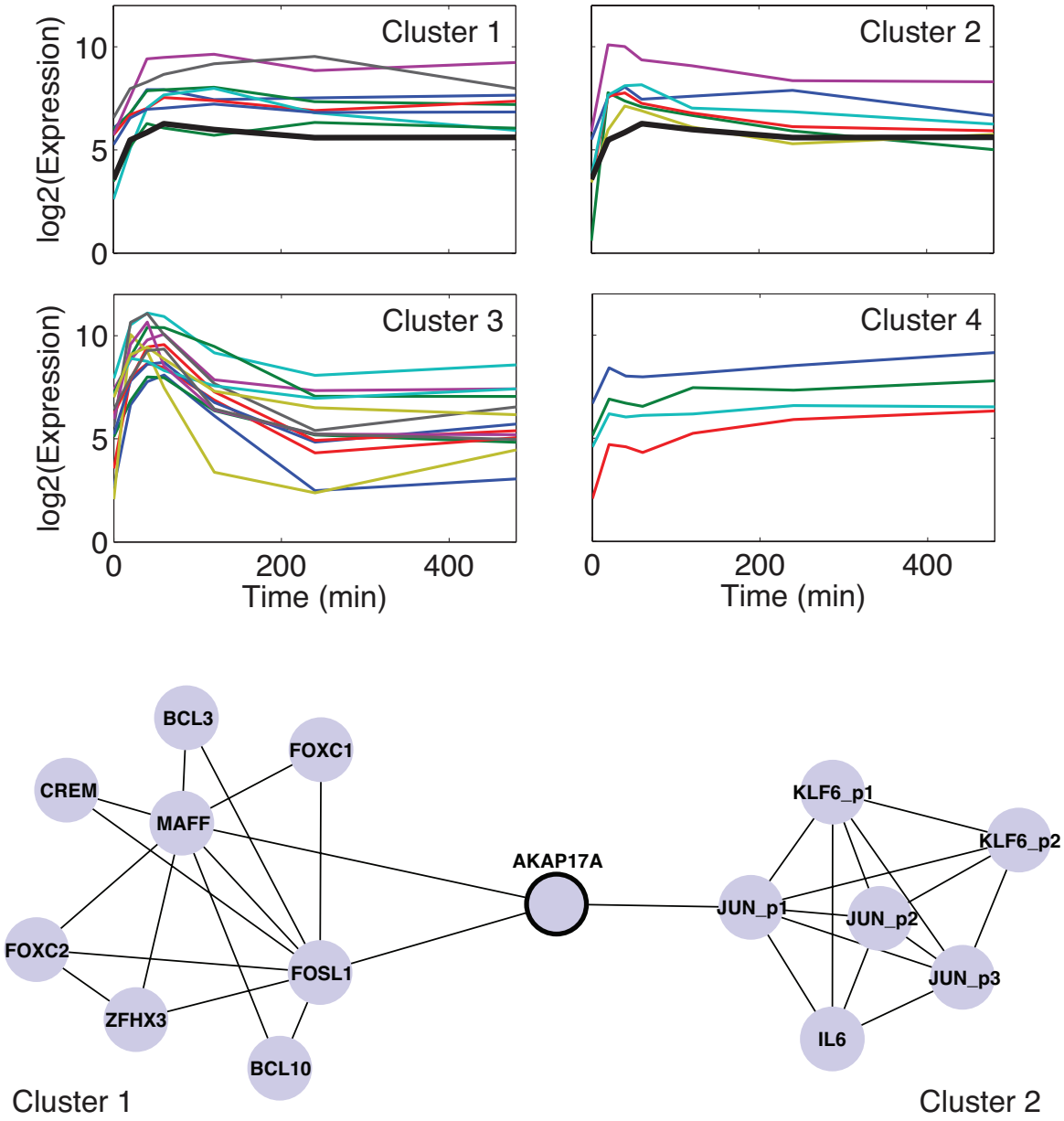

Figure 3.7.: Expression data clustering results

Top two rows: Average representation of dynamics in each robust cluster at K=20, co-clustering frequency >0.65. Emphasized black lines in Clusters 1 and 2 represent dynamics of AKAP17A, which co-clusters robustly with at least one member of both clusters. **Bottom panel:** Graph representation of Clusters 1 and 2. Nodes represent genes. Edges represent co-clustering relationships above the 0.65 robustness cutoff. KLF6_p1 and KLF6_p2 represent separate probe sets hybridizing to KLF6, and similarly for JUN. Node outlined in black represents AKAP17A, belonging partially to both clusters.

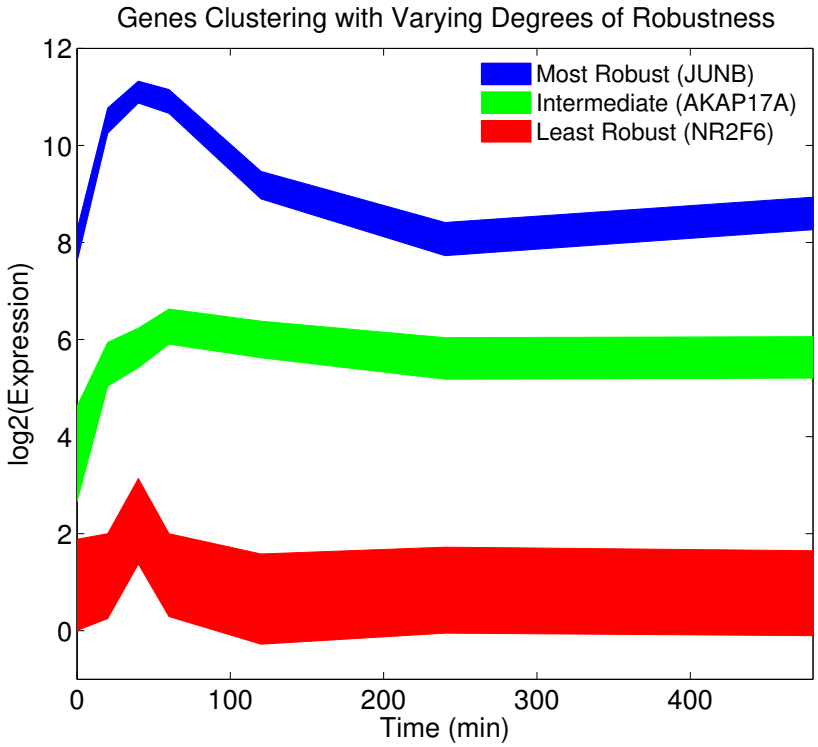

Figure 3.8.: Expression trajectories of genes clustering with varying degrees of robustness

Shaded areas indicate +/- one standard deviation of all resampled vectors for that gene. JUNB clusters most robustly of any gene, AKAP17A is just above the robustness cutoff, and NR2F6 clusters least robustly of any gene. Although relative noise of NR2F6 measurements is higher than that of the other two genes, JUNB and AKAP17A noise is comparable. However, the trajectory of JUNB is more unique, resulting in more robust clustering.

Noise, however, is not the sole determinant of clustering robustness, as previously observed with cluster 5 of the toy example. In ensemble clustering objects belonging to highly separable clusters co-cluster more robustly than objects with comparable noise, but belonging to more densely packed clusters. Figure 3.8 shows trajectories of three genes which cluster with varying degrees of robustness. Despite having comparable amounts of noise, JUNB and AKAP17A do not cluster with the same degree of robustness. JUNB likely clusters substantially more robustly than AKAP17A due to separability properties of their respective

trajectories. While standard deviation of replicate measurements adequately estimates noise, distinguishing clustering properties of JUNB and AKAP17A requires robustness analysis of clustering.

3.6 Conclusions

Unfortunately, experimental measurements are associated with noise, which reduces our confidence in those values. Handling this uncertainty in the process of analyzing large biological datasets by clustering may greatly aid in highlighting those cluster associations which in turn have low or high confidence in light of this noise. Although the incorporation of noise in clustering requires new layers of analysis, compared to not handling noise, these results will ideally help researchers avoid misinterpreting clustering results and allow them to focus on highly probable hypotheses for further study.

We focused this work on those methods that can be used in an algorithm-independent fashion. By analyzing an *in silico* toy dataset and two real biological datasets with very different structures and noise values, we were able to explore how one can incorporate noise in clustering analysis. One of the recurrent lessons across all of these datasets is that the amount of noise alone in a particular measurement does not determine its sensitivity to mis-clustering. The highest sensitivity to noise lies in those regions of high spatial density. Conversely, well partitioned vectors are much less sensitive, indicating noise analysis at the very least can highlight uncertainties in clustering partitions. Additionally, this observation indicates that pre-filtering a dataset to remove observations with a large degree of variation

could remove data from consideration that might be separated well by clustering, despite noise.

The most appropriate method of those presented here is entirely dependent on the dataset being evaluated. The advantage to Method D is that as long as a model for the data and its noise can be assumed, it can be used for single-, low- or missing-replicate data as a way to test the sensitivity of a solution to unobserved noise. However, erroneous noise models could result in over- or under-representation of the sensitivity of the clustering solution for missing- or no-replicate data. Replicate co-clustering or replicate sampling are ways to avoid making assumptions about a particular model of noise, but come at the cost of co-clustering resolution. Since the attainment of these multidimensional datasets typically comes with a large financial or resource burden, it is rare that average-value clustering will be sufficient to handle noise during analysis, given low numbers of replicates. Despite the differences in assumptions and requirements for each method covered here, in toy and phosphoproteomic data, they had fairly rough agreement, and added a great deal of value to the understanding of the stability of a clustering solution, so perhaps the use of any method, despite potential flaws, is still an improvement over not accounting for noise at all in clustering results.

4. High-resolution identification of specificity determining positions in the LacI protein family using ensembles of sub-sampled alignments

This chapter is adapted from the following published manuscript:

> Sloutsky, R. & Naegle, K. M. High-resolution identification of specificity determining positions in the laci protein family using ensembles of sub-sampled alignments. *PLoS One* **11**, e0162579 (2016)

4.1 Abstract

Since the advent of large-scale genomic sequencing, and the consequent availability of large numbers of homologous protein sequences, there has been burgeoning development of methods for extracting functional information from multiple sequence alignments (MSAs). One type of analysis seeks to identify specificity determining positions (SDPs) based on the assumption that such positions are highly conserved within groups of sequences sharing functional specificity, but conserved to different amino acids in different specificity groups. This unsupervised approach to utilizing evolutionary information may elucidate mechanisms of specificity in protein-protein interactions, catalytic activity of enzymes, sensitivity to allosteric regulation, and other types of protein functionality. We present an analysis of SDPs in the LacI family of transcriptional regulators in which we 1) relax the constraint that all specificity groups must contribute to SDP signal, and 2) use a novel approach to robust treatment of sequence alignment uncertainty based on sub-sampling. We find that the vast majority of SDP signal occurs at positions with a conservation pattern that significantly complicates detection by previously described methods. This pattern, which we term "partial SDP", consists of the commonly accepted SDP conservation pattern among a subset of specificity groups and strong degeneracy among the rest. An upshot of this fact is that the SDP complement of every specificity group appears to be unique. Additionally, sub-sampling gives us the ability to assign a confidence interval to the SDP score, as well as increase fidelity, as compared to analysis of a single, comprehensive alignment – the current standard in multiple sequence alignment methodologies.

4.2 Introduction

Rapid advances in DNA sequencing technologies in recent decades have enabled an exponential increase in the number of fully sequenced genomes. Combined with advances in automated gene annotation and functional assignment[227–229], this has resulted in the availability of homologous protein sequences from thousands of species. This abundance of sequence data, in turn, motivated development of numerous computational strategies for inferring functional roles of individual protein residues from the amino acid composition patterns of multiple sequence alignment (MSA) columns.

One such type of analysis seeks to identify residues responsible for specificity differences in families of homologous proteins that share a common function, but differ in substrate, ligand, protein interaction partner, or various other forms of specificity. Starting with the model, first postulated by Susumu Ohno in his seminal book[40], that specificity diversification occurs through gene duplication followed by specialization of each duplicate, the approach further pre-supposes that such specificity-determining positions (SDPs) experience a specific pattern of substitutions following duplication. While positions responsible for their common function remain under constant purifying selection in both duplicates, and positions evolving neutrally diverge through random drift[230], SDPs mutate as the duplicate genes acquire new specificity, then come back under purifying selection once that specificity becomes fixed. Subsequent duplications again relax the purifying selection pressure on SDPs, followed by renewed purifying selection after further specialization. Eventually each specialized gene evolved by repeated duplication gives rise to a set of orthologs – homologs descended from

speciation events – which share both the global function of the protein family and the specificity of their pre-speciation ancestor gene. In the context of SDP identification these are often called specificity groups. Positions responsible for global function remain conserved to the same amino acid across all specificity groups, while neutral positions diverge within each group. SDPs, on the other hand, remain conserved within groups due to purifying selection, but are conserved to different amino acids in each group, as required by its unique specificity. Although the numerous SDP identification algorithms[123,124,126,127,129,131–135,137,139,141,142] differ in their scoring functions, they all reward maximally this "conserved within specificity groups, different between" amino acid composition pattern. Because all methods agree on this, we generically refer to columns with conservation patterns approximating this ideal as having "SDP signal".

Sub-specialization within protein families commonly involves multiple sites in a protein in a combinatorial fashion, possibly including catalytic, allosteric, and interaction sites, as well as other aspects of protein function. In a diverse protein family, each member's specialized function is very unlikely to be determined by the same set of positions. More plausibly, positions acquire and lose specificity roles along different lineages over multiple duplications, resulting in "partial" SDPs which contribute to specialized function in some specificity groups, but not in others. Among the fraction of groups which use a particular position as an SDP, the position should exhibit a conservation pattern consistent with SDP signal. Among remaining groups purifying selection pressure will have been lost, and the position likely reverted to evolving neutrally: diverging through random drift, resulting in

low conservation both within and between groups. In fact, we expect relatively few positions to be under purifying selection in all ortholog sets, with many more positions experiencing a patchwork of purifying selection and neutral evolution across different lineages. If this is the case, one expects to find many positions with a "heterogenous" conservation pattern across ortholog sets: conserved in some sets, degenerate in others. Heterogeneous conservation was previously reported by Casari et al[122] in the Ras/Rab/Rho family, in G2/M and B-type cyclins, and in a small subset of SH2 domains. In larger protein families, at least some heterogeneous positions may contain detectable SDP signal among the specificity groups in which the position is conserved – indicating that this fraction of ortholog sets use the position in a specificity-determining role. Although several methods allow limiting conservation analysis to a subset of input sequences by only considering sequences corresponding to leaves descendant from an internal node in a phylogeny[130,143,231], doing so assumes are relevant signal is contained in this monophyletic subset. However, a partial SDP position that acquired and lost its specificity-determining role multiple times would not have its SDP signal confined to any monophyletic subset of ortholog sets. Identifying SDPs in the context of such non-uniform evolutionary history remains a challenge to understanding specificity in large protein families.

Another, fundamental challenge to all sequence analyses requiring an input MSA, like SDP identification, comes from the uncertainty and imperfect accuracy of the alignment process itself. In all but the most trivial cases, different multiple sequence alignment tools produce differing alignments of the same collection of input sequences. And yet, subsequent

downstream applications treat input alignments as an observation, assuming their correctness[158], even though a number of studies[62,72–74,88,158–161] have demonstrated sensitivity of downstream applications to alignment variability. To make matters worse, two recent studies demonstrated strong positive correlation between the number of aligned sequences and the overall amount of alignment error for every tested alignment tool[62,63]. Furthermore, after repeatedly aligning a constant subset of sequences with different collections of additional homologs, Sievers *et al.*[63] found that the embedded alignment of the constant subset was affected by the variable additional sequences – illustrating sensitivity of pairwise alignments embedded in an MSA to the total number and context of aligned sequences. Although a number of approaches for identification and removal of alignment columns with the most uncertainty have been developed[162–166], simply removing columns is of limited utility for column-wise analyses like SDP identification. Therefore, using all available sequence data, in a manner robust to alignment uncertainty and inaccuracy, is a second challenge in SDP analysis of large protein families.

In this work we identify numerous partial SDPs in the LacI family of bacterial transcriptional regulators, previously analyzed by multiple SDP identification methods[126,129,132,139,144]. LacI family members vary in their DNA binding specificity, allosteric regulator identity and promiscuity, and even regulatory logic – with some members dissociating from DNA upon binding their regulators and others requiring their regulator to bind DNA[232]. Since the LacI family contains at least 34, possibly as many as 45 members, each represented by a set of orthologs from numerous bacterial species[233], it also poses the challenge of robustly

analyzing MSAs of large collections of homologs. To address this challenge we employ sub-sampling to generate an ensemble of LacI MSAs, taking advantage of a large amount of sequence data, while aligning relatively few sequences at any one time. We extend an existing SDP identification method, *GroupSim*[142], in order to account for partial SDPs and to calculate group-specific scores – allowing us to determine whether a position is an SDP for some groups, but not for others. We find support for partial SDP in the physical interactions of corresponding side chains in solved structures of LacI and its homologs. In comparing group-specific SDP scores in our work with two other methods, SDPPred[126,129] and Speer[139,144], we find that group-specific scoring identifies many positions that cannot be detected by existing methods and highlights where these methods are likely making false positive SDP calls for subsets of specificity groups. Consistent with our expectation for a protein family with complex specificity, and in contrast to SDPPred, Speer, and *GroupSim*, SDP complements identified by our group-specific method vary dramatically between family members. The resulting aggregate analysis is robust to alignment uncertainty and inaccuracy, with individual sequence position results demonstrating a wide range of sensitivity to alignment variation. Our sub-sampling approach constitutes a general framework for robust treatment of any SDP method and, more generally, of any computational analysis of multiple sequence alignments.

4.3 Results

We assembled a pool of 1814 unique sequences covering 20 members of the LacI protein family, each represented by a set of orthologs, consisting of between 28 and 192 sequences, from different bacterial species. Since a multiple sequence alignment (MSA) of this many sequences will suffer from significantly higher error[63], we opted to align a subset of 200 sequences randomly sampled from the pool. To create sufficient sampling of the full sequence space, we repeated this sub-sampling and alignment 5000 times to form an ensemble of MSAs. In order to merge analysis results across the ensemble, we included a reference sequence in each set, for a total of 201 sequences in every alignment. Results were aggregated by reference sequence position and are referenced that way throughout the text. To avoid bias the reference sequence was withheld from analysis and only the 200 sampled sequences were used. Six separate ensembles were generated, each with a respective reference sequence representing one of the six family members with a solved structure: AscG, CcpA, FruR, LacI, PurR, and TreR. Positions in reference sequences were independently mapped to each other with a structural alignment, allowing us to compare results for structurally homologous sequence positions in different family members. Because results from all six ensembles were highly similar, we report results based on the LacI reference sequence (LacI of *Escherichia coli*, UniProt accession P03023), unless otherwise specified.

Our ensemble approach allowed us to quantify the variability column-wise metrics experience as a result of differences in alignment inputs and specific errors, which will be high-

lighted throughout the remaining results. In short, by using the average SDP score across the ensemble, the result becomes more robust to uncertainty in the alignment process.

4.3.1 Detection of SDP signal at heterogeneously conserved positions

We assume each member of the LacI family has unique specificity and, therefore, we treat sets of family member orthologs as specificity groups for the purposes of SDP analysis. This assumption is predicated on the fact that paralogs with identical function are extremely rare. Instead, when the two copies of a gene resulting from a duplication event fail to evolve functional differences, one copy tends to become a pseudogene[234].

Throughout the text "ortholog set" and "specificity group" both refer to the collection of orthologs of a family member protein from different bacterial species. "Family member" is also used to refer broadly to all orthologs of a protein.

Relationships between conservation, agreement, and SDP signal

We find it useful to represent alignment columns as points projected into a two-dimensional space – where the first dimension is the variable quantifying net amino acid conservation within specificity groups (group-wise conservation) and the second dimension is the variable quantifying net agreement between amino acid compositions of groups (between-group agreement) (Fig. 4.1). This projection is conceptually similar to the two entropies projection, total column entropy and sum of entropies of each specificity group, used by Ye *et al.*[134]. We

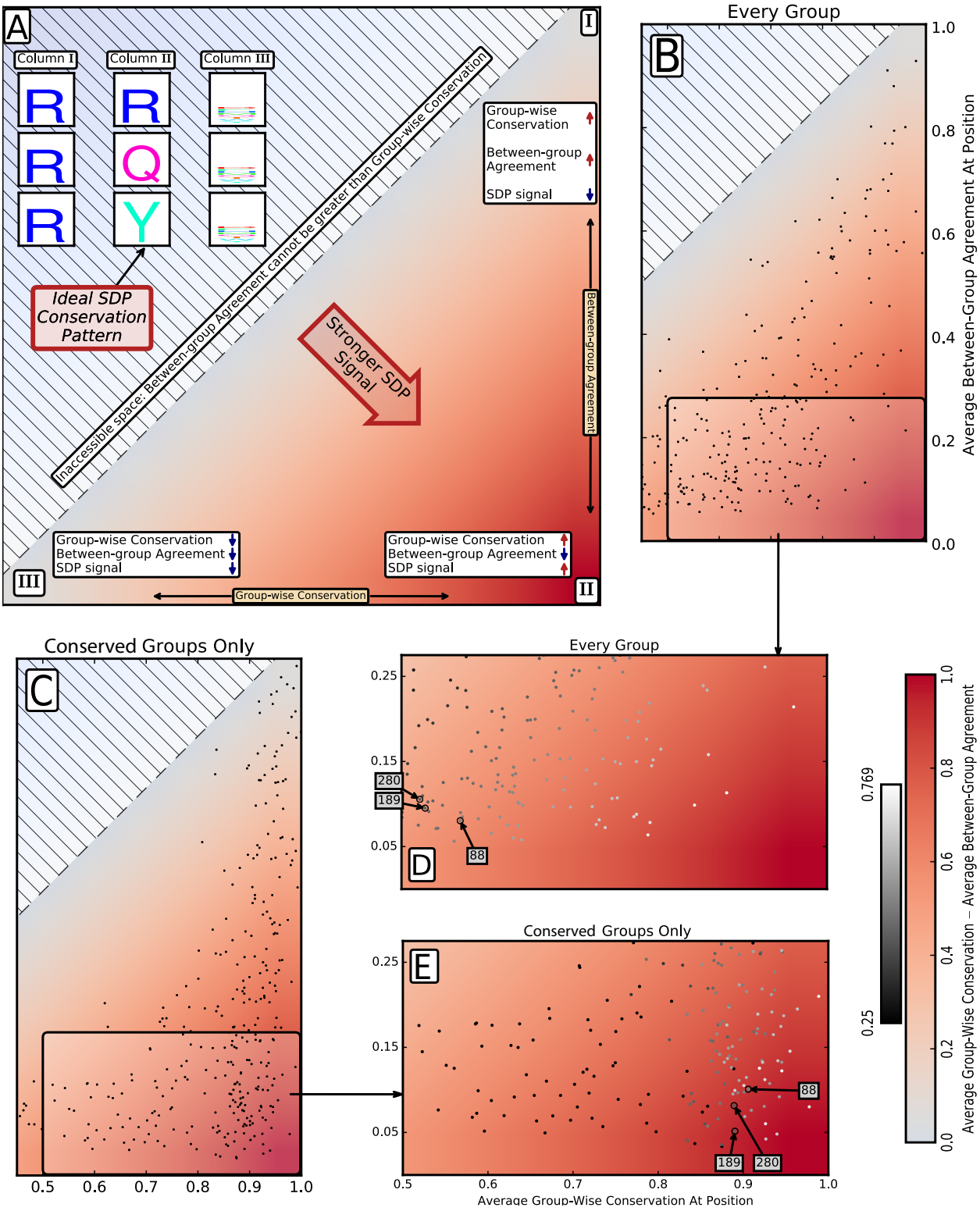

Figure 4.1.: Projection into conservation-agreement space

Figure 4.1.:

In every panel, the color gradient represents strength of SDP signal, as quantified by average group-wise conservation minus average between-group agreement. Dark red (bottom right quadrant) represents maximal SDP signal. (**A**) Projections of hypothetical alignment columns for illustration: Column II has maximal SDP signal, while columns I and III have low signal. (**B**,**C**,**D**,**E**) Projections of LacI reference sequence positions with group-wise conservation and between-group agreement computed either (**B**,**D**) over every specificity group or (**C**,**E**) over conserved groups only, where group conservation is >0.6. (**D**) Points corresponding to LacI positions are colored in grayscale corresponding to the red color gradient of (**B**). (**E**) Points are positioned according their SDP signal calculated over conserved groups only, but using the grayscale of (**D**) for illustration of the shift individual sequence positions undergo as a result of the altered scoring scheme of (**C**).

then calculate SDP signal according to the method in *GroupSim*[142], defined as the difference between group-wise conservation and between-group agreement.

Fig. 4.1(A) illustrates the fundamental relationships between group-wise conservation, between-group agreement, and SDP signal in the two-dimensional space. Conservation is maximal and agreement is minimal when every specificity group is strictly conserved to a different amino acid – the ideal SDP pattern (Fig. 4.1, Column II). Regardless of its specific scoring function, every SDP identification method awards its maximum score to alignment columns with this pattern. Similarly, every method awards a low SDP score to columns where every group is conserved to the same amino acid (Fig. 4.1, Column I): high conservation and high agreement, since it is proposed such positions cannot determine specificity differences. Low SDP signal is also assigned when most groups are degenerate (Fig. 4.1, Column III) – i.e. conservation is a mandatory component of SDP signal. The consequence of this requirement is that the larger the fraction of degenerate groups, the more the SDP signal degrades.

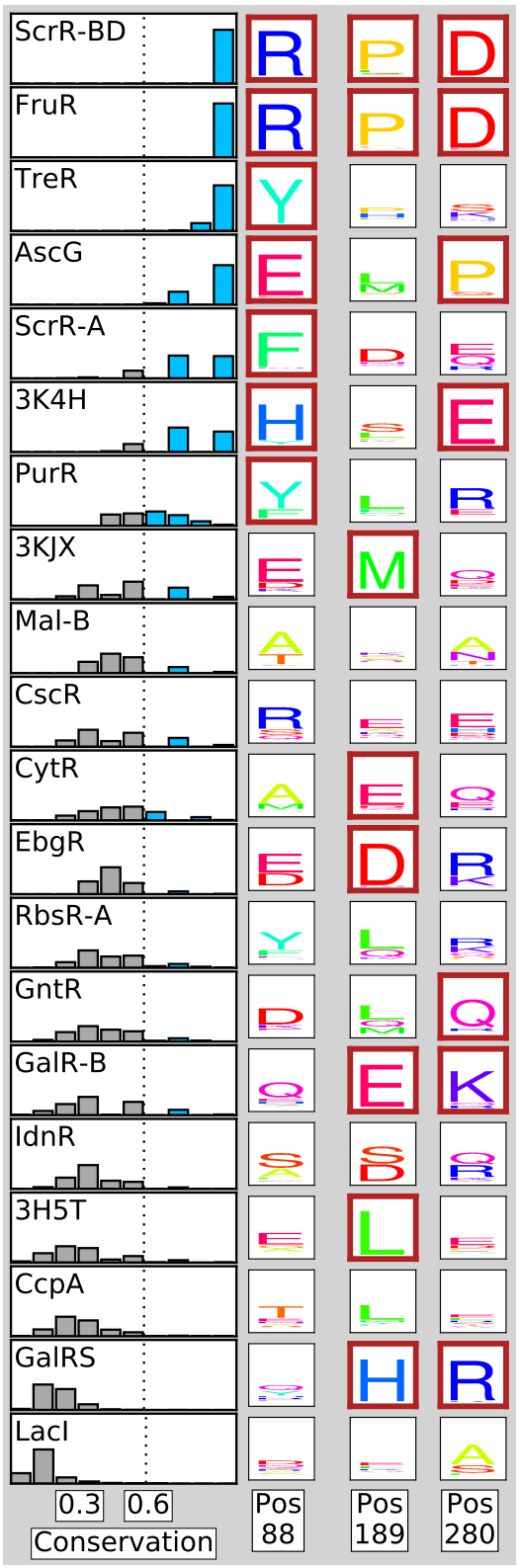

Figure 4.2.: Amino acid composition at heterogeneously conserved positions

Figure 4.2.:

Historgrams at left show group conservation distributions at position 88 over the MSA ensemble for each family member. The dotted line indicates threshold for "conserved" designation, separating high conservation in blue from low conservation in gray. Amino acid content of each of the 20 ortholog sets is represented by sequence logos for three positions that demonstrate heterogeneous conservation. Rows correspond to LacI family members. Sequence logos for ortholog sets with average group conservation above the conservation cutoff are outlined in maroon.

Quantifying group conservation and between-group agreement across the ensemble

Analysis of any property of an alignment column can be extended across an ensemble of alignments. A benefit of the ensemble approach is the ability to explore the distribution of a property over collections of input sequences. For example, Fig. 4.2 demonstrates the distributions of conservation within each of the 20 ortholog sets representing 20 LacI family members for a single position (LacI reference position 88). In almost all cases there is variability in this calculation (the only exceptions are the strictly conserved scR-BD and fruR families). By taking the average value for conservation and agreement, we are, ideally, creating robustness to the variability of these metrics as a function of alignment.

In order to establish a metric for high conservation within a specificity group across the ensemble, we call a group conserved if its average conservation score is 0.6 or greater. For a group of eight sequences, this threshold corresponds to six or more amino acids being identical. In Fig. 4.2 ScrR-BD and FruR orthologs are most conserved at reference position 88, with conservation of 1.0 in every ensemble alignment, while LacI orthologs are least conserved, and consistently so across the ensemble. We define a column as heterogeneously

conserved, or heterogeneous, when specificity groups in it span conservation extremes: at least six groups have conservation greater than 0.6 and at least six others have conservation less than 0.5.

Conservation heterogeneity is pervasive

A third of LacI reference sequence positions (124 of 360) exhibit heterogeneous group conservation. We represent both conservation and amino acid content over the ensemble at three positions with heterogeneous conservation (positions 88, 189, and 280) by sequence logos[235,236] (Fig. 4.2). The subset of conserved groups varies dramatically from one heterogeneous position to another. On average, a specificity group is conserved at only 55 of 124 positions, and no group is conserved at more than 82 positions, suggesting that purifying selection pressure is acting on a unique subset of positions in each ortholog set.

Noise from degenerate groups masks strong SDP signal at some positions

Plotting reference sequence positions in conservation-agreement space illustrates the impact of conservation heterogeneity on SDP signal across all positions (Fig. 4.1(B,D)). Since so much of the LacI sequence is heterogeneously conserved across family members, the area of strongest SDP signal (bottom right quadrant Fig. 4.1(B)) is relatively unpopulated. Noise from degenerate groups hampers detection of SDP signal among conserved groups by lowering group-wise conservation and making the position as a whole indistinguishable from positions with uniformly lower conservation across all groups.

In Fig. 4.1(C,E) amino acid positions are re-plotted according to a calculation including only the subset of groups identified as being conserved (group conservation ≥ 0.6) at a position. This process ideally removes the noise contributed by degenerate groups normally included in traditional SDP calculations. Naturally, when only conserved groups are considered, group-wise conservation increases for all positions, except those at which every group is conserved – resulting in a shift of all positions to the right. However, comparing Fig. 4.1(D) and Fig. 4.1(E) demonstrates that this shift is far from homogeneous. Two color gradients are used in order to compare the original, all group calculation, with the calculation based only on the subset of groups that demonstrate conservation. In Fig. 4.1(E), where positions are plotted by conserved groups only, the area of strongest SDP signal is populated by a mixture of points having variable SDP signal in the original scoring scheme. For example, positions 88, 189, and 280, whose group amino acid composition is shown in Fig. 4.2, are three of the biggest beneficiaries of the modified scoring scheme. While removing noise from degenerate groups increases SDP signal overall, individual positions still vary in the strength of signal among their conserved groups. Based on this analysis, we incorporated this filter into a high-resolution SDP metric.

4.3.2 Detection of SDP signal in individual specificity groups

As expected for a diverse protein family, the vast majority of noise-filtered SDP signal in the LacI family is contributed by positions with high heterogeneity of conservation, i.e. positions at which a subset of specificity groups are degenerate and another subset of groups

are conserved. We propose a simple method for identifying partial SDPs by evaluating SDP signal in a group-specific manner. Here, we compare the results of this approach to three existing methods, SDPPred, Speer, and *GroupSim*, which – like all existing methods – assign a single score to every specificity group in an alignment column. Our results suggest that the standard approach can produce both false positives and false negatives as a result of heterogeneous conservation across groups.

A group-specific SDP score

We compute a modified *GroupSim* score, filtered for noise from degenerate groups by only including conserved groups, where group conservation ≥ 0.6, in the score calculation. We refer to these conserved groups as "support" groups, since only these groups can provide support for an SDP call. For each specificity group in an alignment column, we then modulate the score by a weight that accounts for the evidence of the position's importance to this group, based on the group's conservation. Specifically this is calculated according to the following:

$$W_{group} \times (\text{ group-wise conservation over support } - \text{ between-group agreement over support })$$ (4.1)

where

$$W_{group} = \begin{cases} 1 & \text{if group} \in \text{support} \\ \text{group conservation} & \text{otherwise} \end{cases} \tag{4.2}$$

Averaging this score over the ensemble of MSAs accounts for heterogeneity in a group's conservation. Groups conserved at a position in every ensemble MSA receive a higher score than groups conserved at the position in a fraction of MSAs. The outcome of this approach is an individualized score for every specificity group (Figs. 4.3 and 4.4).

SDP signal is highly variable across specificity groups

We compare results of our group-specific method with the *GroupSim* method, on which our method is based, and with two other existing methods, SDPPred and Speer, for the 20 highest scoring LacI sequence positions as judged by either of the latter two methods (Fig. 4.3). SDPPred, Speer, and *GroupSim* scores for a position apply to every specificity group. Overlap between SDPPred, Speer, and *GroupSim* is high – 16 positions are among the top 20 for all three methods – confirming that different methods generally detect the same SDP signal. However, group-specific scoring demonstrates that SDP signal, defined as being in the top 7.5% of all group-specific scores, is never uniformly high across all specificity groups. SDP signal detected by SDPPred, Speer, or *GroupSim* is supported by, on average, only 12 of 20 specificity groups. Therefore, the group-specific scoring scheme is able to identify groups with low SDP signal due to low conservation. Given that conservation within

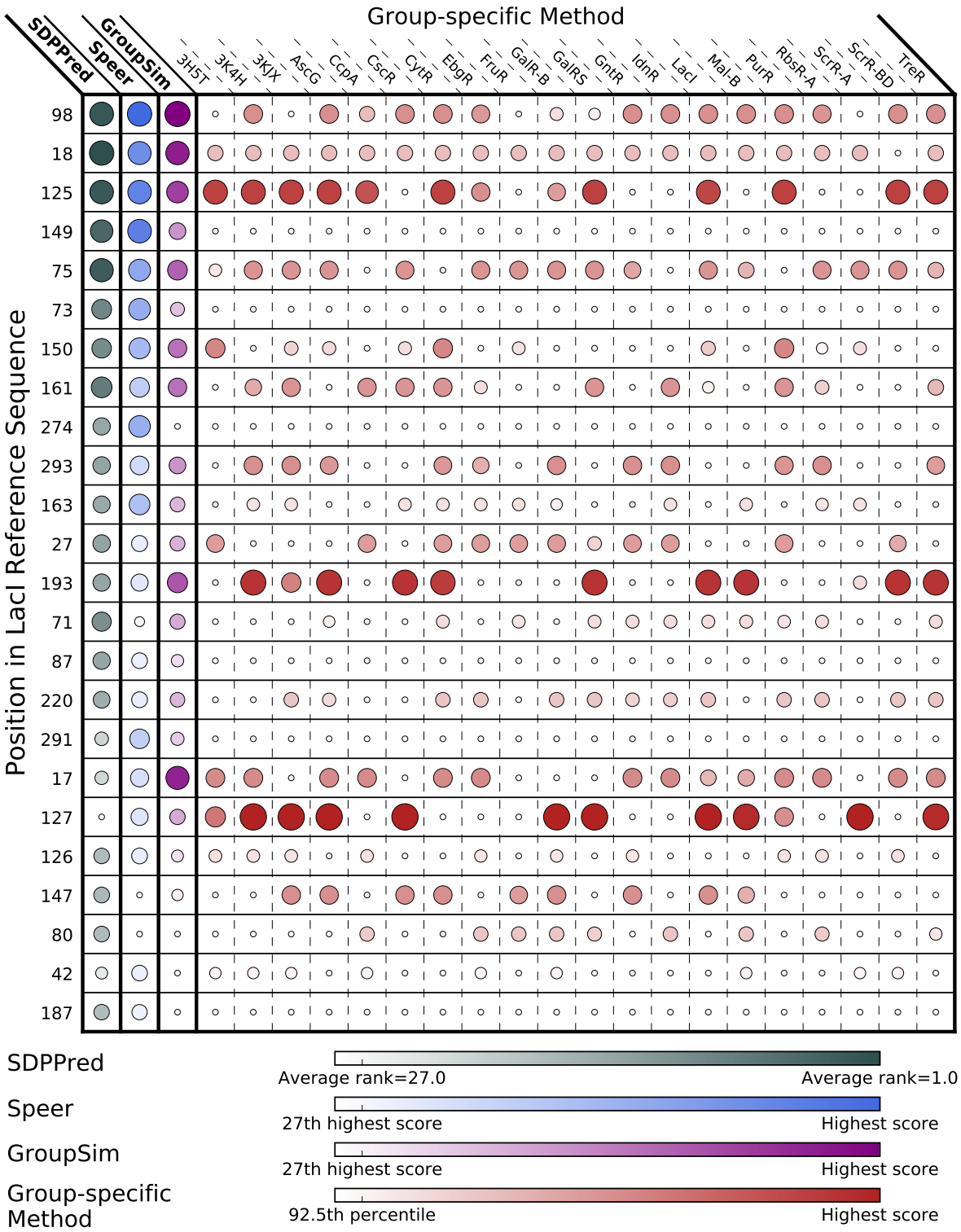

Figure 4.3.: SDP results for the highest scoring positions by SDPPred and Speer

Figure 4.3.:

Each position receiving a top-20 score from at least one of the comparative methods, SDPPred and Speer, are shown. Ensemble score for SDPPred is the average ranking. Ensemble score for Speer is the average z-score. See *Methods* for details on SDPPred and Speer ensemble averages. Position scores are shown for SDPPred, Speer, and *GroupSim*. Group-specific scores for each specificity group at the corresponding position are also shown. Marker size and color correspond to score according to color bars. Note that top 7.5% of scores make up the vast majority of color scale for each method. For column-wise scoring methods the 27th highest score corresponds to the 92.5th percentile, since $27 \div 360 = 0.075$, or 7.5%.

a specificity group is a requisite for hypothetical importance in a specificity determining role, it is likely that traditional methods are overcalling SDPs at these positions for those groups and a group-specific scoring scheme rectifies this.

Our method identified 15 additional LacI positions with strong SDP signal, where at least one group's score is in the top 5% of all group-specific scores (Fig. 4.4). All of these positions score outside the top 20 for both SDPPred and Speer, likely due to the fact that, on average, only 7.4 of 20 groups have detectable signal in this set. Position 29 scores 11th highest with *GroupSim*, underscoring the modest differences between existing methods, but the remaining 14 positions in Fig. 4.4 score outside of the top 20 for *GroupSim* as well. Noise from numerous degenerate groups masks the SDP signal at these positions when SDP is calculated as a property of all groups. Our group-specific method detects partial SDPs even when the signal is present in a small fraction of specificity groups.

Figs. 4.3 and 4.4 offer a striking illustration of the complexity of specificity encoding in LacI family proteins. Every single position with detectable signal is a partial SDP to some extent, and no two positions appear to have signal in the same subset of family members. There are some positions (62, 81, 128, 189, 191, 196, and 277) that additionally highlight the

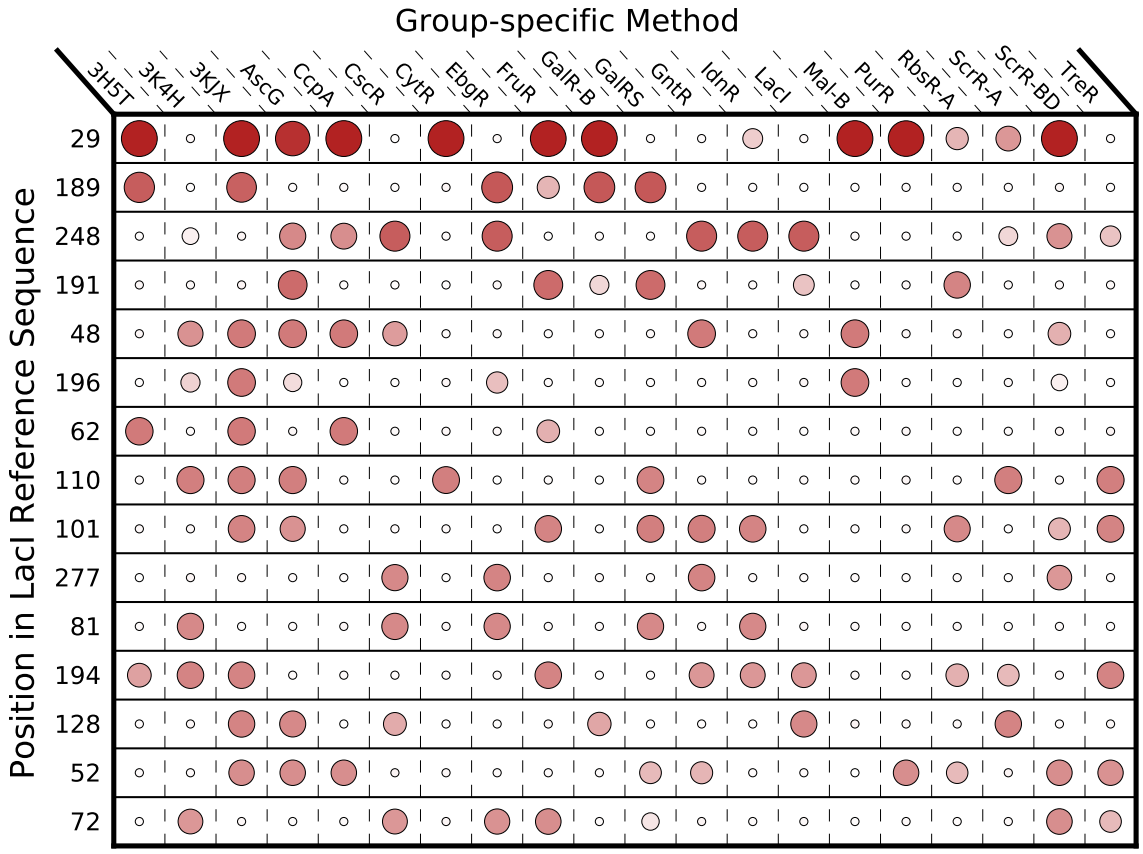

Figure 4.4.: Group-specific SDP signal undetected by SDPPred or Speer
Marker size and color corresponds to group-specific score according to color bar in Fig. 4.3

sensitivity of SDP analysis to available sequence data, since all of these positions would have failed to have high SDP signal, should the latter three ortholog sets not been included in this analysis. Non-inclusion of a group could easily occur if there was low representation of these orthologs in currently sequenced species. This highlights the sensitivity of SDP analysis to input and importance of using all available data.

A subset of positions score among the top 20 with either SDPPred or Speer, but outside of the top 7.5% for our group-specific method: 149, 73, 274, 87, 291, and 187. Of these, 149,

73, 87, and 291, but not 274 or 187 score in the top 7.5% for *GroupSim* (Fig. 4.3), though *GroupSim* scores each position lower than SDPPred or Speer. The fact that *GroupSim* ranks these positions higher than the group-specific method is misleading: group-specific scores for conserved groups at these positions are actually *higher* than *GroupSim* scores (because the two methods use the same scoring function, scores can be compared directly). However, because their SDP signal is selectively boosted by the noise filtering in our method, positions in Fig. 4.4 crowd positions 149, 73, 87, and 291 outside of the top 7.5%. The group-specific method prioritizes positions that are very different from those prioritized by existing methods.

From exploring the similarities among positions ranked higher by other methods than by our group-specific method (Supporting Information), a clear pattern emerges: strength of SDP signal detected by any method falls as the fraction of groups conserved to the same amino acid increases, resulting in greater between-group agreement. While SDPPred, Speer, and *GroupSim* detect SDP signal at some or all of these positions, none of the four methods detect signal at positions 22 or 25 (Supporting Information). In addition, it appears that the *GroupSim* scoring function penalizes between-group agreement somewhat more severely than those of SDPPred and Speer, explaining why each position with this pattern is ranked lower by *GroupSim* (Fig. 4.3). Detecting SDP signal in conservation patterns like positions 22 and 25, at which a large fraction of groups are conserved to the same amino acid, presents a considerable challenge to all SDP identification methods.

4.3.3 Structural organization of group-specific SDPs

The position of a residue in the 3-dimensional structure of a protein can provide clues to its role in protein function and specificity. Therefore, we explored SDP positions for those families where structures are available. We mapped positions scoring in the top 5% of all group-specific SDP scores onto family members with solved structures (Figs. 4.5, 4.6, Supporting Information, Supporting Information, Supporting Information, and Supporting Information). Based on group-specific scores, this results in a unique structural collection of SDPs for each family member.

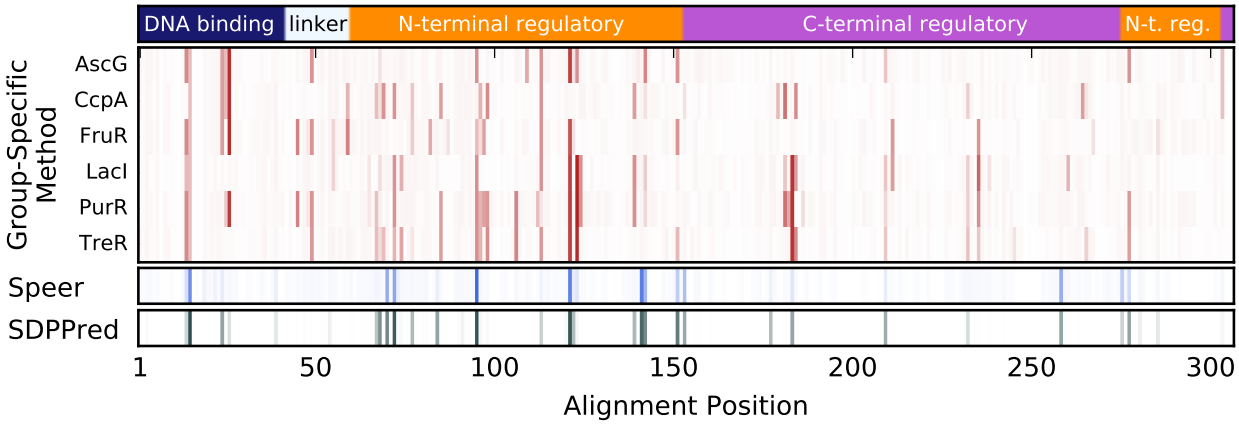

Figure 4.5.: SDP scores mapped onto reference structural alignment

Locations of alignment positions in structural features are indicated in the top track. The allosteric site is located at the interface of N-terminal and C-terminal regulatory sub-domains, each of which is split into two linear segments of the polypeptide chain, as indicated. Heatmap colors correspond to group-specific scores for indicated specificity groups and whole-position Speer and SDPPred scores, according to the color bars in Fig. 4.3.

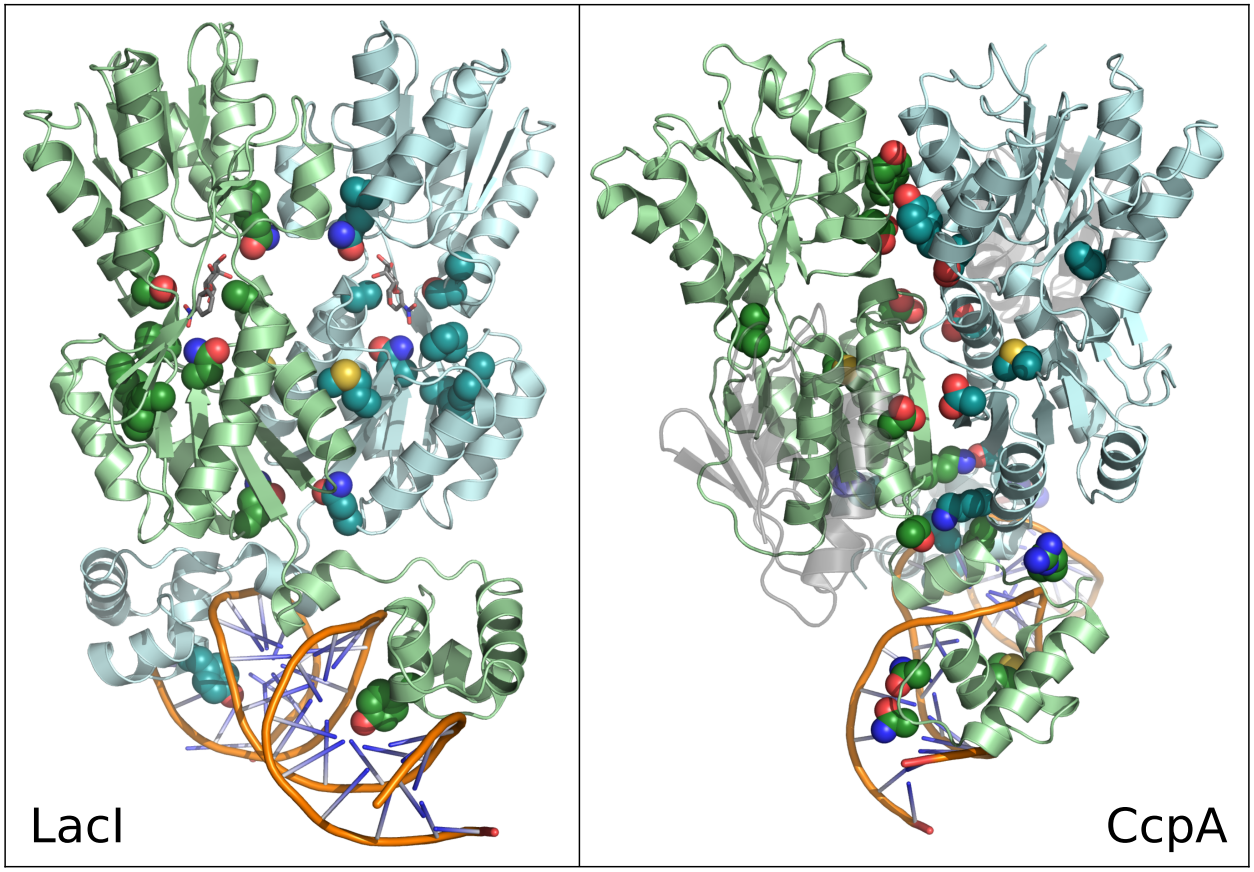

Figure 4.6.: Structural distribution of SDP complements of LacI and CcpA

LacI (left) and CcpA (right) SDPs scoring in top 5% of all group-specific SDP highlighted on LacI (2pe5) and CcpA (3oqo) structures. Each protein is shown as a homo-dimer complexed with DNA, with one monomer shown in blue and the other in green. SDP side chains shown in space-filling representation in color matching their monomer. LacI ligand and CcpA binding partner protein shown in gray. CcpA binding partner is semi-transparent.

SDP complements of family members have unique structural organization

In order to compare SDPs in their sequence alignments to structure, we created a structural sequence alignment of the AscG, CcpA, FruR, LacI, PurR, and TreR reference sequences and compared this to SDP scores (Fig. 4.5). There is, overall, substantially more SDP signal in the N-terminal half of the alignment, corresponding to the helix-turn-helix DNA binding

subdomain, the inter-domain linker, and the N-terminal regulatory subdomain. Together these account for DNA binding functionality and, most likely, the conformational transition induced by binding and dissociation of the allosteric regulator. In addition, several SDPs in the C-terminal regulatory subdomain are in the allosteric site located at the interface of N-terminal and C-terminal regulatory subdomains. By comparison, the remainder of the C-terminal sub-domain is relatively devoid of SDP signal.

In order to locate the positions of top-scoring SDPs within the 3-dimensional structure, we mapped SDPs onto the structures of two family members, LacI and CcpA (Fig. 4.6). SDP complements of LacI and CcpA identified by the group-specific method clearly have different spacial organization. LacI SDPs cluster near the allosteric binding site and in the adjacent protein core region, where they are likely participate in ligand-induced conformational changes. Only a single DNA contacting residue has strong SDP signal in LacI, although additional DNA contacting residues have an SDP-like conservation pattern, impossible to detect by any of the four methods due to high between-group agreement (as discussed earlier). On the other hand, CcpA SDPs cluster almost exclusively at monomer-monomer and protein-DNA interfaces, with three SDPs contacting DNA. The prevalence of positions at the interface between monomers suggests CcpA diverged from the rest of the LacI family in some functional aspect of dimerization.

Structural maps of AscG, FruR, PurR, and TreR SDP complements are shown in Supporting Information through Supporting Information. The comprehensive mapping of SDP signal onto available structures suggests that family members diverged through specializa-

tion in varying aspects of function, as indicated by clustering of SDPs at different locations in the protein.

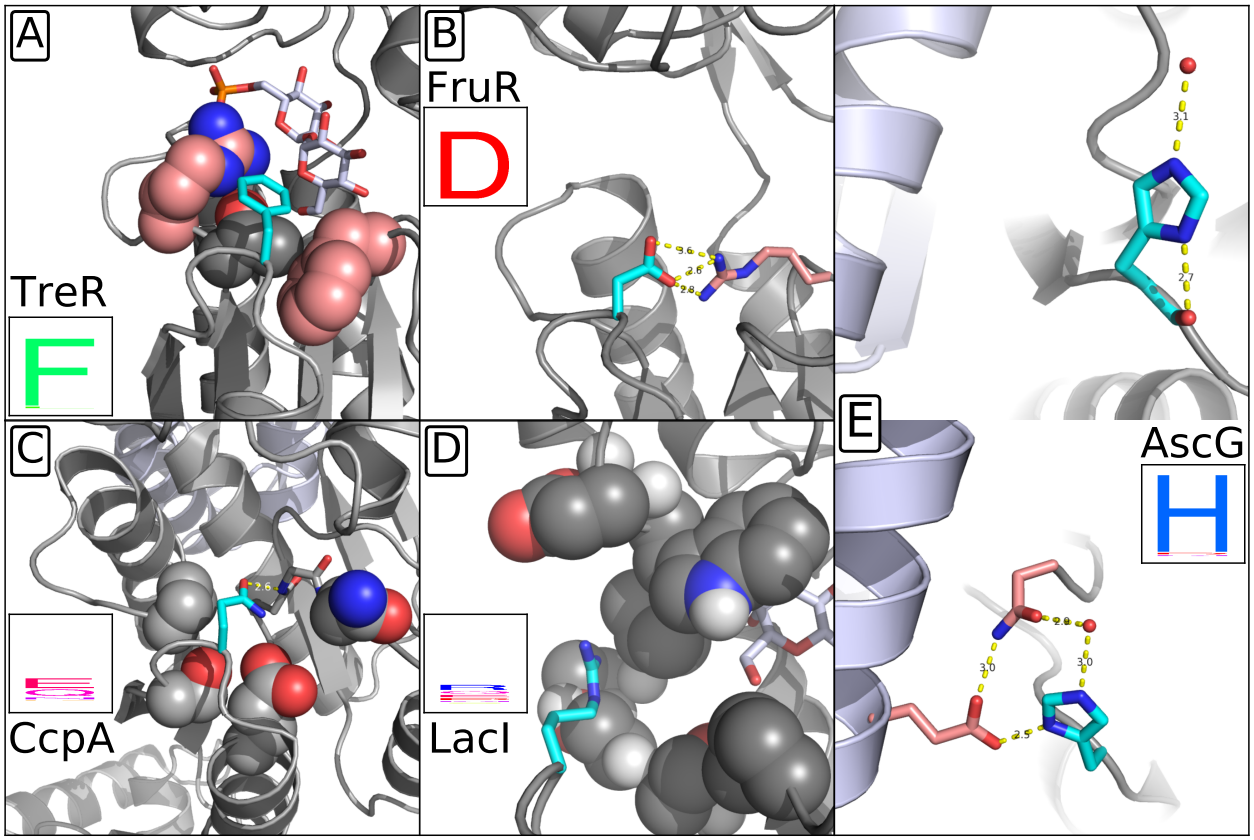

Figure 4.7.: Structural evidence of partial SDP at LacI position 101

Interactions of TreR (**A**), FruR (**B**), CcpA (**C**), LacI (**D**), and AscG (**E**) positions corresponding to LacI position 101, according to the structural alignment. The side chain at the position homologous to LacI 101 is shown in light blue. Side chains at neighboring positions are shown in salmon, if those positions are SDPs, and in gray otherwise. Amino acid composition of the ortholog set is represented by sequence logo. Packing interaction of TreR F102 with F127 and hydrogen bonding interaction of FruR D101 with R149 are highly specific. CcpA Q101 and LacI R101 do not form specific interactions, although CcpA Q101 does participate in a single hydrogen bond. Glutamic acid and asparagine, capable of making the same interaction, also occur among CcpA orthologs. LacI R101 is exposed to solvent, and several other polar amino acids occur at the position. AscG H101 participates in two different interactions. (**E**), **top:** hydrogen bonding with cis-monomer backbone (gray) and coordinated water molecule (red dot). (**E**), **bottom:** hydrogen bond network with cis-monomeric N68, trans-monomeric E88 (light violet backbone), and another coordinated water.

Structural evidence that an SDP is used by only a fraction of family members

In comparison to other methods, our method has increased the total number of positions with significant SDP signal. Additionally, our group-specific scoring scheme uses group conservation to identify subsets of specificity groups that are most and least likely to use the position as a specificity determinant. We illustrate our method's ability to identify these subsets by highlighting the structural roles of residues at a position where our method identified a partial SDP – position 101 in the LacI reference sequence (Fig. 4.7). These residues (TreR F102, FruR D101, CcpA Q101, LacI R101, and AscG H101) are homologous to each other, according to the structural alignment, and correspond to position 101 of the LacI reference sequence in our analysis. In TreR, FruR, and AscG this position is conserved to three unique amino acids and accordingly, all three received very high group-specific SDP scores. In their respective structures all three participate in highly specific hydrophobic packing (TreR) or hydrogen bonding (FruR, AscG) interactions which cannot be satisfied by other amino acids. In contrast, in LacI and CcpA this position is degenerate and receives low group-specific scores. Accordingly, R101 of LacI has no obvious interactions with either the nearby ligand or any neighboring residues, none of which are SDPs. Since the position is exposed to solvent, theoretically any polar residue should be tolerated. This is borne out by the range of amino acids occurring at this position in LacI orthologs. In CcpA Q101 forms a single hydrogen bond with a nearby backbone nitrogen atom. Again, none of the neighboring positions are SDPs. Asparagine and glutamic acid, both capable of forming the same hydrogen bond, are present at this position in other CcpA orthologs. AscG H101

presents a particularly interesting case study for this partial SDP position. Histidine is strictly conserved in AscG orthologs and the group-specific SDP score is high. In Fig. 4.7(E), the two H101 residues in an AscG dimer participate in two different interactions - one trans and one cis - neither of which alone appears to strictly require histidine. However, only histidine can satisfy both interactions simultaneously, consistent with its conservation among AscG orthologs.

These structural observations support the hypothesis that position 101 contributes to specificities of TreR, FruR, and AscG, but not of LacI or CcpA. For AscG, although neither H101 interaction alone provides evidence supporting SDP, the two taken together are consistent with the SDP call. This example demonstrates the usefulness in group-specific scoring, which detected both the importance of position 101 to specificity groups in which it is conserved and its lack of a specific role in specificity groups in which it is degenerate.

4.3.4 Sensitivity of ensemble SDP scores to alignment uncertainty

Results reported so far were obtained from an ensemble of MSAs. In order to compare ensemble results to the traditional single-MSA approach, we created a single, "comprehensive" alignment of all 1814 sequences and scored it with our group-specific SDP method. Even for SDP signal in the top 1%, when groups are most conserved, comprehensive alignment scores are often outliers with respect to score distributions over the ensemble (Fig. 4.8). Consistent agreement between the average score from the ensemble and the score from the comprehen-

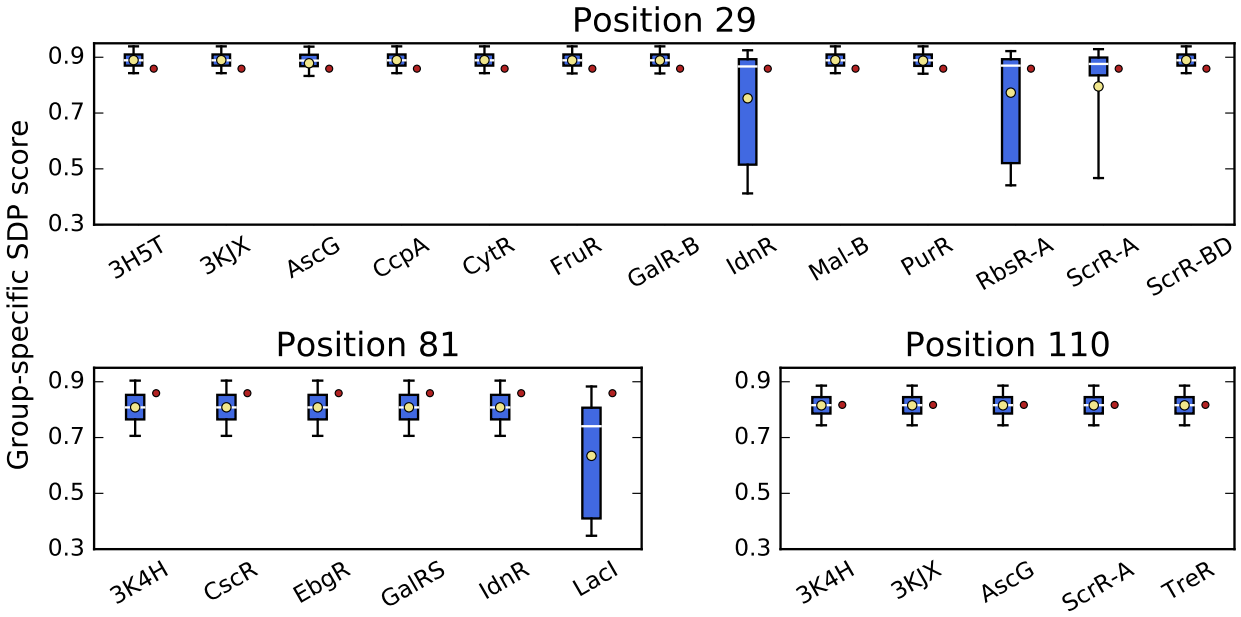

Figure 4.8.: SDP score distributions vs comprehensive alignment scores

Score distributions and the comprehensive alignment scores for specificity groups with a score falling in the top 1% are plotted for positions 29, 81, and 110. Score distributions shown as box plots, with medians indicated by white lines and means indicated by yellow dots. Boxes cover middle two quartiles of score distributions, while whiskers cover middle 95%. Comprehensive alignment scores shown as red dots. These can fall below (position 29), above (position 81), or within (position 110) the middle two quartiles of the ensemble distributions. Some ortholog sets (IdnR, RbsR-A, ScrR-A at position 29, LacI at position 81) can be substantially more sensitive to alignment variability than other ortholog sets at the same position. This fact is reflected in their ensemble score (distribution average - yellow dot), but not in the comprehensive alignment score.

sive alignment, such as seen at position 110, is rare. More often the comprehensive alignment score falls in the tails of ensemble score distributions, such as seen at positions 29 and 81.

In most cases ensemble score distributions are symmetric, as indicated by similar mean and median values of the distribution. Symmetric score distributions with low variance suggest that the same amino acid nearly always aligned to this reference sequence position for all orthologs in that specificity group. The ensemble method identifies specificity groups for which conservation varied dramatically between alignments, indicating greater uncertainty

in the alignment of those orthologs at that position – e.g. IdnR and RbsR-A at position 29 – and penalizes the specificity group for this uncertainty with a lower ensemble score (ensemble distribution average). The comprehensive alignment approach cannot account for different degrees of alignment uncertainty between specificity groups: all groups receive a single score.

4.4 Discussion

In this work we demonstrated that a substantial fraction of positions in the LacI family are heterogeneously conserved – i.e. only a fraction of family members are highly conserved, while a comparable fraction are highly degenerate. In order to accurately identify the specificity determinants among positions with this conservation pattern, we implemented a scoring approach in which we 1) boost SDP signal-to-noise ratio by considering only the specificity groups that are conserved at a position and 2) modulate the score in a group-specific manner – based on each group's degree of conservation. The paralog-specific collections of specificity determining residues identified using our method cluster on their representative protein structures in configurations that are consistent with our understanding of the functional specialization of those proteins. Importantly, the modulation of the score appears consistent with the importance of the corresponding residue, given its physical interactions. Our scoring method avoids spurious SDP identification for family members in which a position is degenerate and detects "hidden" SDPs used by a small fraction of family members.

In the course of our investigation, we encountered a conservation pattern that occurred at positions ranked significantly lower by our method than by SDPPred, Speer, or even *GroupSim*, which uses the same scoring function as our method. The pattern is characterized by conservation of a large fraction of specificity groups to the same amino acid, consistent with specialization of the common ancestor of those groups, followed by maintenance of the same functional role through the more recent duplications that gave rise to present day specificity groups. For example, at position 22, 15 of the 20 groups are conserved to arginine, while the remaining groups are conserved to one of four other amino acids. While SDPPred and Speer do tolerate a marginally greater amount of between-group agreement than the *GroupSim* scoring function, their, and *GroupSim*'s ability to rank these positions higher than our method is a side-effect of their failure to detect SDP signal at a number of positions identified by our method (Fig. 4.3), rather than a strength. In addition, they too fail to identify positions with conservation patterns like that of positions 22 and 25 (Supporting Information) as SDPs.

Several SDP methods can simultaneously identify SDPs and optimal specificity groups [127,132,137,143] by grouping sequences so that total SDP signal across all alignment columns is maximized. However, as Supporting Information illustrates, such columns often have mutually exclusive optimal sequence groupings, which further conflict with many partial SDPs identified in this work. These observations suggest that further development of SDP identification methods may be required to identify SDPs with high between-group agreement.

In this work we also tackled the common challenge of MSA-based computational analyses that arises from uncertainty of the alignment process due to both sensitivity to the input collection of sequences and to alignment error. This concern is particularly acute when analyzing large collections of sequences, because overall alignment error increases rapidly with the number of aligned sequences. We avoided making large alignments, while still taking advantage of all available sequence data, by building and analyzing ensembles of sub-sampled MSAs. Using an ensemble average improves the robustness of any metric computed on a sequence alignment and allows for the detection of regions in the alignment that may be especially prone to error. We believe this robust approach can be generalized to any analysis that requires an MSA input.

Whether "specificity determining position" is a biologically meaningful designation remains an open question. Highly targeted experiments are necessary to demonstrate this functional role: for example, by demonstrating that substituting the amino acids at these positions with the amino acids present at the homologous positions in a paralog is sufficient to switch the functional specialization of the protein to that of the paralog. The partial SDPs identified in this work, together with the ortholog sets in which these positions are conserved, will significantly reduce the number of candidates for mutation that must be considered by experimentalists when investigating specialization in the LacI family.

4.5 Methods

4.5.1 Generation of MSA ensembles

We downloaded all protein sequences from the LacI family resource AlloRep[233] and supplemented each ortholog set with sequences from EnsemblBacteria release 26[237]. A supplemental sequence was added to an ortholog set, if: 1) it had 35% or greater identity to each ortholog in the set and 2) its lowest identity to any ortholog in the set was higher than its identity to any other sequence in the pool. We then dropped from our analysis any ortholog set containing fewer than 20 sequences in order to ensure adequate statistical coverage. The final sequence pool contains 1814 sequences split among 20 ortholog sets ranging from 28 ortholog sequences (IdnR) to 192 ortholog sequences (CcpA).

The subsamples of 200 sequences were sampled from each ortholog set according to its frequency in the full sequence set. We required a minimum allocation of eight sequences to avoid small number effects and limited the maximum to 13 sequences per ortholog set. This sampling procedure was repeated 5000 times. Each 200 sequence sample was combined with a reference sequence and the 201 sequences were aligned using MAFFT's L-INS-i (most accurate) protocol[94,238]. In addition to the LacI reference sequence, AscG (P24242), FruR (W8ZE48), PurR (X7PN48), and TreR (P36673) of *Escherichia coli* and CcpA (P25144) of *Bacillus subtilis* were used as reference sequences.

4.5.2 SDP scoring

Pairwise comparisons between sequence positions, $comp(s_1, s_2)$, were made using the identity matrix which had previously produced the most accurate results with both XDet[135] and *GroupSim*[142] SDP identification methods. Conservation within a specificity group was defined as the average of pairwise comparisons between all sequences in the group:

$$\langle\ comp(s_1, s_2)\ \rangle_{\{(s_1,s_2)\ \forall\ s_1 \in group,\ \forall\ s_2 \in group\ |\ s_1 \neq s_2\}} \tag{4.3}$$

For an alignment column, group-wise conservation was defined as the average of each group's conservation:

$$\left\langle\ \langle\ comp(s_1, s_2)\ \rangle_{\{(s_1,s_2)\ \forall\ s_1 \in group,\ \forall\ s_2 \in group\ |\ s_1 \neq s_2\}}\ \right\rangle_{groups} \tag{4.4}$$

and between-group agreement was defined as the average pairwise sequence comparison between sequences belonging to different groups, averaged over all pairs of groups:

$$\left\langle\ \langle\ comp(s_1, s_2)\ \rangle_{\{(s_1,s_2)\ \forall\ s_1 \in g_1,\ \forall\ s_2 \in g_2\}}\ \right\rangle_{\{(g_1,g_2)\ \forall\ g_1 \in groups,\ \forall\ g_2 \in groups\ |\ g_1 \neq g_2\}} \tag{4.5}$$

5000 alignments from the LacI ensemble were scored with SDPPred[126,129], accessed via its web interface at http://bioinf.fbb.msu.ru/SDPpred/, and Speer[139,144], downloaded from ftp://ftp.ncbi.nih.gov/pub/SPEER/ and run locally.

SDPPred produces a ranking of positions with statistically significant scores for every alignment. The number of ranked positions varies from alignment to alignment, and there is no clear way to rank positions without statistically significant scores. For each position in the LacI reference sequence we averaged its rank across all ensemble MSAs to generate an ensemble score. All positions not ranked by SDPPred for a particular MSA received the next rank after the last explicitly ranked position: e.g., if SDPPred ranked 20 positions, every unranked position received rank 21 for averaging purposes. Because of this, ensemble scores for SDPPred are not discriminatory beyond, roughly, rank 30.

Speer produces several scores, including a z-score based on the mean and variance of scores for each position in an alignment. We averaged the z-scores of each LacI position over the MSA ensemble to produce an ensemble Speer score.

4.5.3 Structural mapping of SDPs

We aligned representative protein structures for each reference sequence with MUSTANG[239] to produce an independent structural alignment of the reference sequences. Structures 3dbi (AscG), 3oqo (CcpA), 2iks (FruR), 1jwl, 1tlf, 2pe5 (LacI), 1jft, 2pua (PurR), and 4xxh (TreR) were aligned. Structures with multiple ligands were used for LacI and PurR. The DNA binding subdomain and inter-domain linker segments were not included in any structures of AscG, FruR, or TreR. In order to obtain a complete mapping, full reference sequences were aligned to the structural alignment using MAFFT's seeded alignment option.

4.5.4 Implementation

Group-specific scoring code is available at http://naegle.wustl.edu/software.

4.6 Supporting Information

Figure 4.9.: S1 Fig

Figure 4.9.:

Amino acid content at SDPs with excess between-group agreement Amino acid content of each of 20 ortholog sets, represented by sequence logos, at positions with an SDP-like group-wise conservation pattern. Between-group agreement increases from left to right. Position 18 receives high scores from SDPPred, Speer, and the group-specific scoring method. Positions 149 through 187 are detected, with progressively lower scores, by at least one of SDPPred and Speer, but not by the group-specific method. Positions 25 and 22 are not detected by any method.

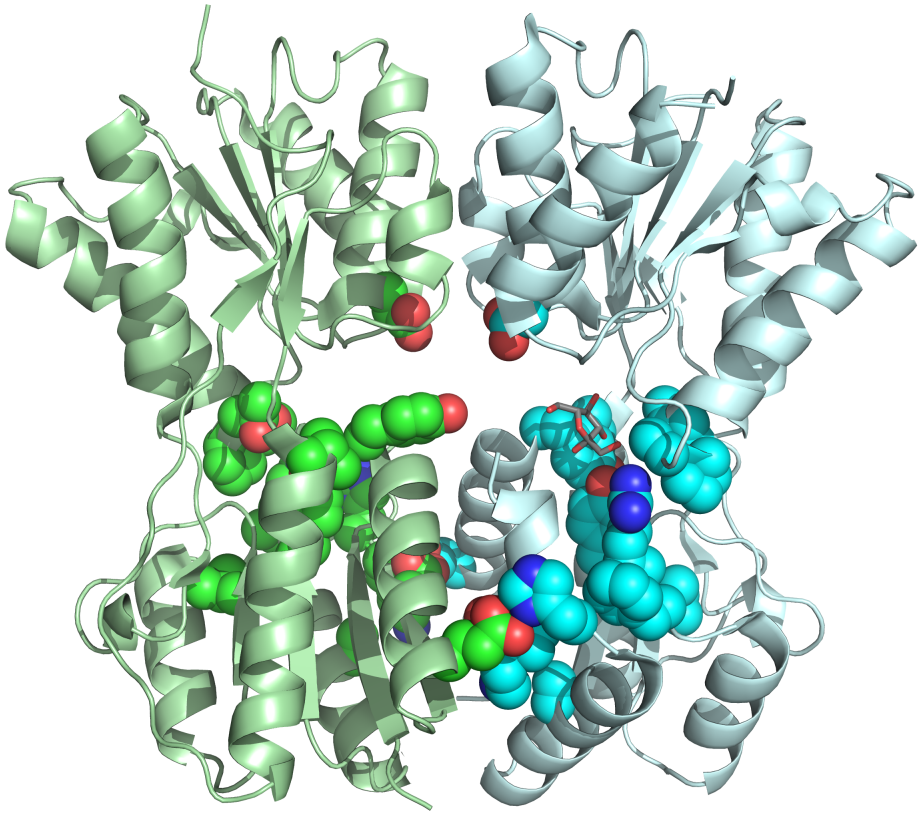

Figure 4.10.: S2 Fig

SDP complement of AscG. SDPs mapped onto structure 3brq and highlighted in space-filling representation. Structure only contains N- and C-terminal regulatory subdomains.

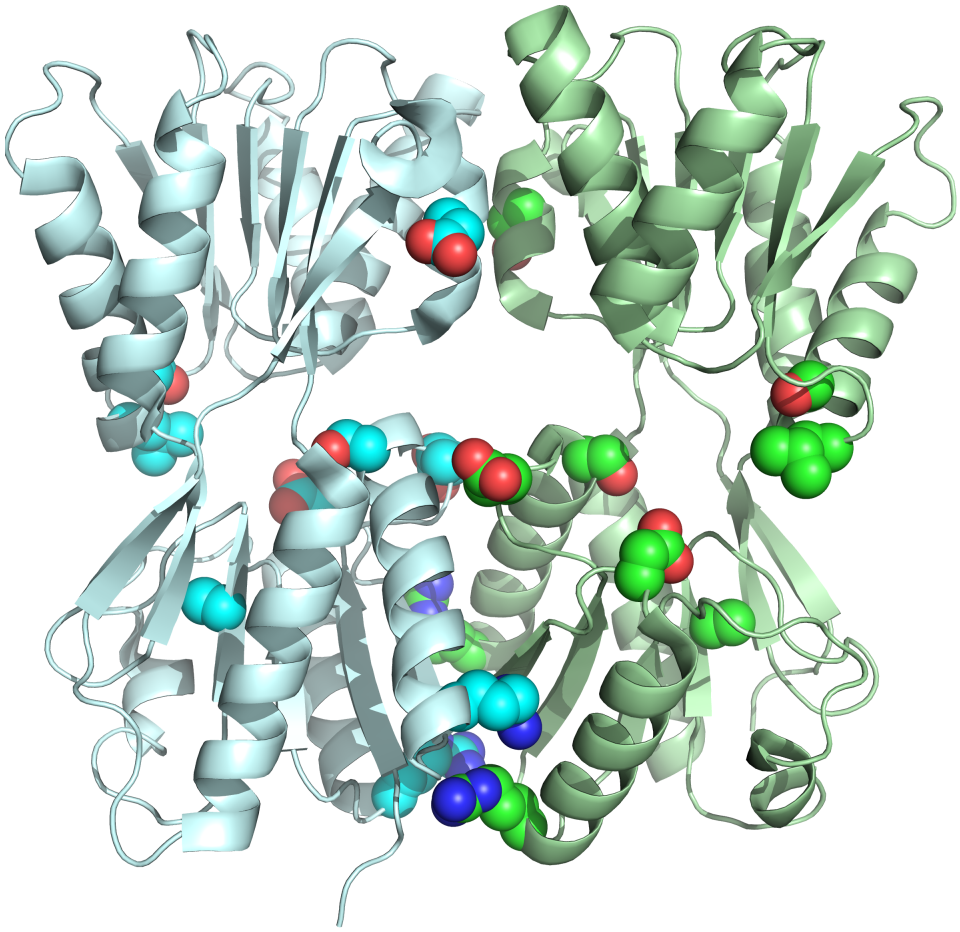

Figure 4.11.: S3 Fig

SDP complement of FruR. SDPs mapped onto structure 2iks and highlighted in space-filling representation. Structure only contains N- and C-terminal regulatory subdomains.

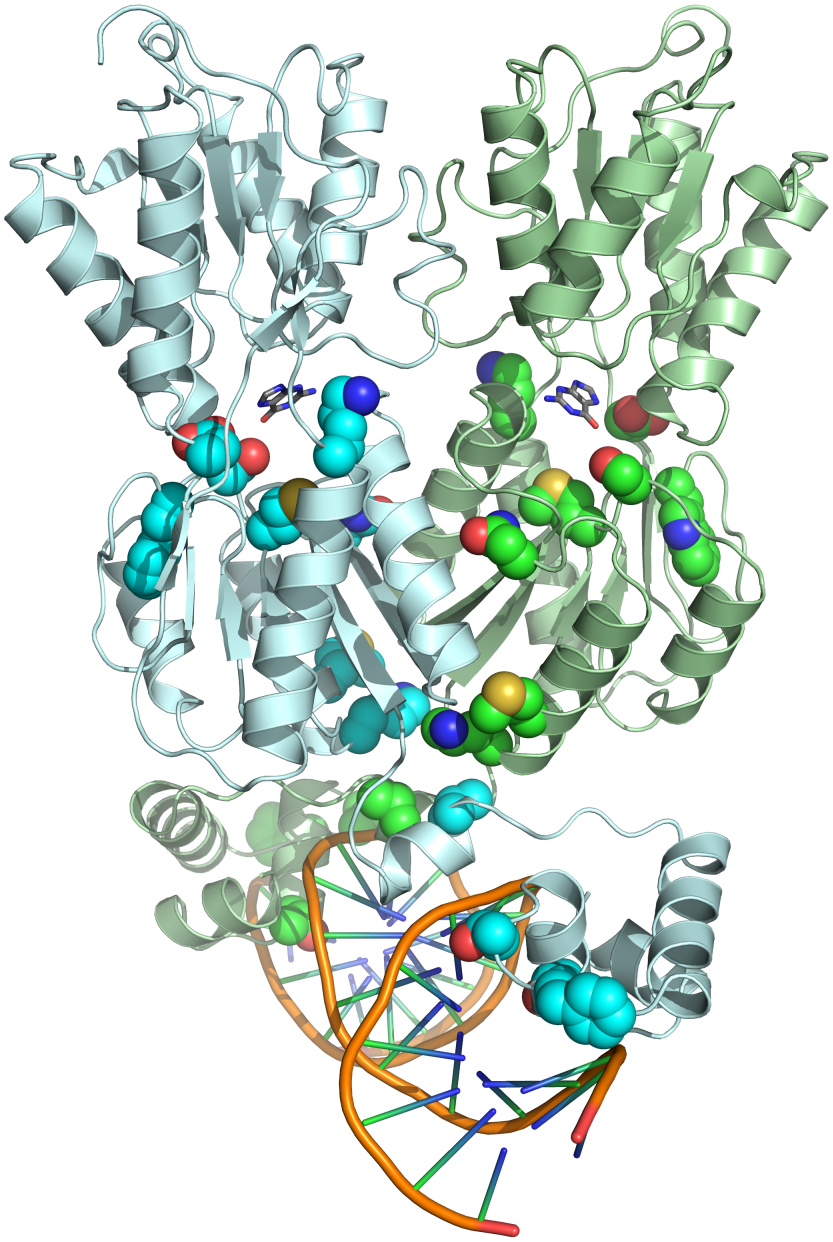

Figure 4.12.: S4 Fig

SDP complement of PurR. SDPs mapped onto structure 2puc and highlighted in space-filling representation.

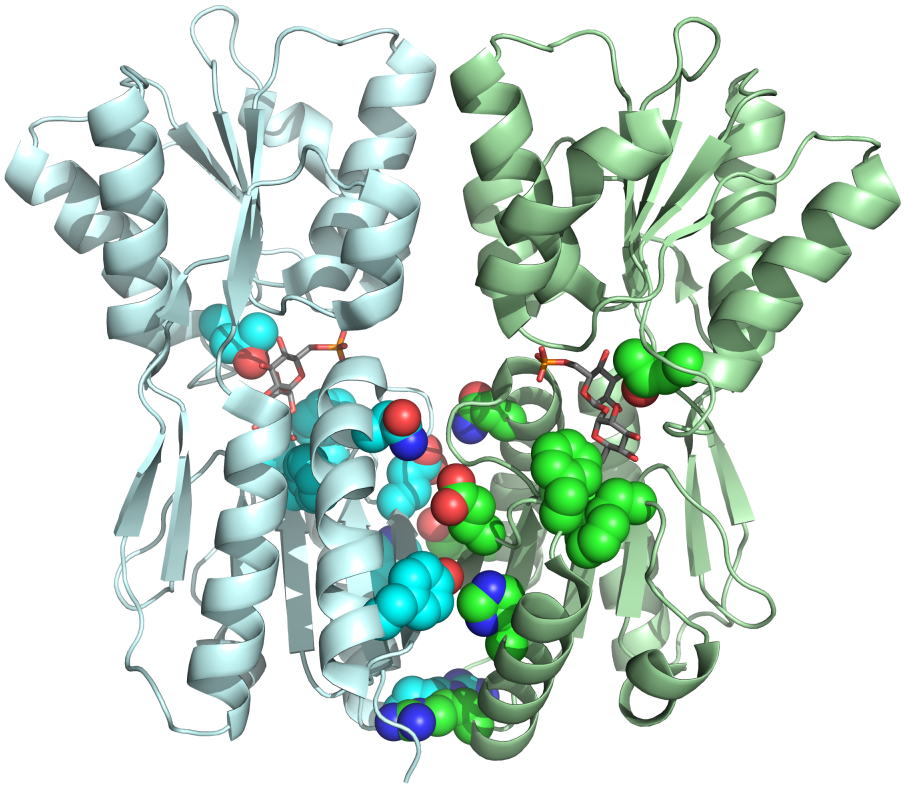

Figure 4.13.: S5 Fig

SDP complement of TreR. SDPs mapped onto structure 4xxh and highlighted in space-filling representation. Structure only contains N- and C-terminal regulatory subdomains.

4.7 Acknowledgments

We wish to thank Barak Cohen, Gary Stormo, Justin Fay, Jim Havranek, and Tom Ronan for helpful discussions. We also wish to thank two anonymous reviewers for their comments on the manuscript.

5. Accuracy through Subsampling of Protein EvolutioN: Analyzing and reconstructing protein divergence using an ensemble approach

This chapter is adapted from a manuscript in submission, co-authored by myself and Kristen Naegle.

5.1 Abstract

Mapping the history of gene duplications which gave rise to a protein family encoded in a genome (a set of paralogs) can be critical to understanding how those proteins function in their host cells today. However, since each member of a family is recapitulated in the genomes of related species (a set of orthologs), selection of sequences to be included in the history reconstruction is non-trivial. Reconstruction is extremely sensitive to the choice of sequences, which is deeply problematic given no mechanism exists for assessing the accuracy of individual reconstructions. Here, we capitalize on the variability of phylogenetic tree reconstruction to selected input sequences, by subsampling from the available ortholog sequences of a protein family to create an ensemble of trees, which explores the space of plausible tree topologies. We hypothesize that the most consistent topological features across an ensemble are more likely to be true and propose a tree reconstruction algorithm (ASPEN) based on this hypothesis. We simulate 600 protein families over known phylogenies, with varying branch lengths, and compare the accuracy of ASPEN reconstructions to those of traditional phylogeny inference methods. We find that ASPEN trees are more accurate than trees reconstructed traditionally. Additionally, we develop an observable metric calculated form subsampling, reconstruction Precision, for assessing the likely accuracy of a traditional, single-alignment all-sequence reconstruction of the divergence history for a set of paralogs. Together these findings suggest that an ensemble of imperfect reconstructions can provide more accurate insight than any individual reconstruction.

5.2 Introduction

Protein families grow in size and diversity through duplication of genes encoding existing family members followed by functional divergence of the duplicates[40,42]. Immediately following a gene duplication event the affected genome contains two identical copies of the duplicated gene. Because the genes are redundant, relaxed purifying selection allows mutations to accumulate rapidly. Since the added energy cost of expressing identical products from redundant loci confers a selective disadvantage, mutations resulting in loss of functionality by one of the copies are typically favored by selection. However, the rapid accumulation of mutations can also result in partial or complete functional divergence between the two copies. This may create a selective advantage due to increased functional repertoire through neofunctionalization, greater efficiency and control through sub-functionalization, and possibly resistance to deleterious mutations through vestigial functional overlap (functional moonlighting)[41,234,240], leading to retention of both diverged copies (paralogs). After subsequent speciation events give rise to diverged genomes (species), each of those genomes contains a gene descended through speciations from each paralog in the ancestral genome (Figure 5.1). These genes are orthologs characterized by a "same gene, different genome" relationship. Ortholog sets are related to each other as paralogs, since their respective Most Recent Common Ancestors (MRCAs) were the original paralogs in the ancestral genome. The genome of each species encodes a paralog gene belonging to each ortholog set.

Reconstructing the divergence history (topology) of a protein or protein domain family is crucial to understanding the proteins' (protein domains') function(s) and evolution. In

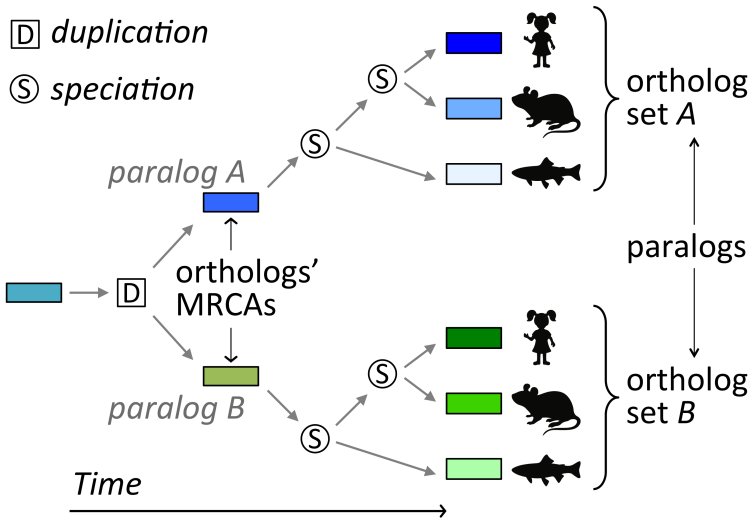

Figure 5.1.: A hypothetical protein divergence history

Two paralogs emerge after a duplication event and are passed along through subsequent speciation events. If no additional duplication events occurred, paralogs *A* and *B* existed at one time as Most Recent Common Ancestors (MRCAs) of two ortholog sets and exist today in the genomes of species emerged through the series of speciations. Each ortholog set can be thought of as representing its MRCA's paralog.

addition to facilitating powerful *in silico* analyses[66,68,130,241–245], reconstructions of paralog divergence guide experimental design and data interpretation[246–251]. Accordingly, divergence reconstructions for well-studied protein domain families[29,30,252] have been relied upon extensively by the scientific community. Because such reconstructions are created from single sequence alignments, they ignore the great deal of uncertainty in topology reconstruction under equally valid alignment representations of input sequence data.

Divergence topology reconstruction is extremely sensitive to the input alignment. For example, the same sequences aligned by different algorithms[62,73,74,158,253] or using different guide trees[84] yield different topology reconstructions. So does reversing input sequences prior to alignment[157,166], or removing less than 0.1% of columns from an alignment containing

over 600,000 columns[254]. For paralog divergence topologies, another source of uncertainty likely to influence reconstruction is the set of orthologs selected to represent each paralog. Because duplications usually predate numerous speciation events, they tend to correspond to deep internal nodes – nodes with many descendant leaves – in full phylogenies of protein families. MRCAs of ortholgs descend from duplications (Figure 5.1), meaning every ortholog descended from each MRCA is also descended from the duplication. Deep internal nodes tend to be most sensitive to perturbations of the input alignment[255]. Unfortunately, since the true history of protein divergence is hidden from us in time, we have no way of knowing which divergence topologies are more accurate, given the equal validity of input alignments.

Although traditional tree reconstruction produces phylogenies – topologies parametrized with branch lengths reflecting extent of divergence – we disregard the branch lengths here to focus on the topologies alone. In traditional inference topologies and branch lengths are inferred jointly, alternating between topology modifications and branch length optimization in the case of statistical (Maximum Likelihood and Bayesian) methods. Because the likelihood function is evaluated many times for each proposed topology, and topology space is almost unfathomably large, statistical methods can suffer extremely long run times on large sequence collections. However, if accurate candidate topologies can be identified by other means, the computational cost of optimizing branch lengths for individual topologies is nearly trivial, while optimization for multiple topologies is embarrassingly parallel. Our approach permits separating topology reconstruction from branch length optimization.

Furthermore, we focus on reconstructing only the hardest topology nodes – the deep internal nodes corresponding to protein or domain paralog divergence. We treat MRCAs of ortholog sets as leaves in our reconstructions and disregard ortholog divergence, which overwhelmingly recapitulates the species tree. Species divergence is reconstructed more accurately by other approaches[254,255]. Instead, we capitalize on the variance in reconstructed topologies under changes in ortholog representation of paralogs to separate topological features we believe to be supported by phylogenetic signal from spurious ones we believe to result from noise. We hypothesize that features observed more frequently under ortholog resampling are more likely to reflect signal and, therefore, be more accurate, than less frequently observed ones. We explore the relationship between accuracy and variability in reconstructing paralog divergence topologies and propose a metric for assessing the likely accuracy of a single-alignment reconstruction for a given protein family. We then present ASPEN, a topology reconstruction algorithm that integrates over the uncertainty of single alignment reconstructions to build and rank trees according to observations across reconstructions from many equally valid alignments. ASPEN produces more accurate topologies than traditional reconstructions from single, all-sequence alignments.

5.3 Experimental framework for reconstruction analysis

We generated test sequence data by simulating evolution of protein families instead of using natural protein sequences for two previously noted reasons[74]. First, simulating evolution over known phylogenies allowed us make a quantitative assessment of reconstruction

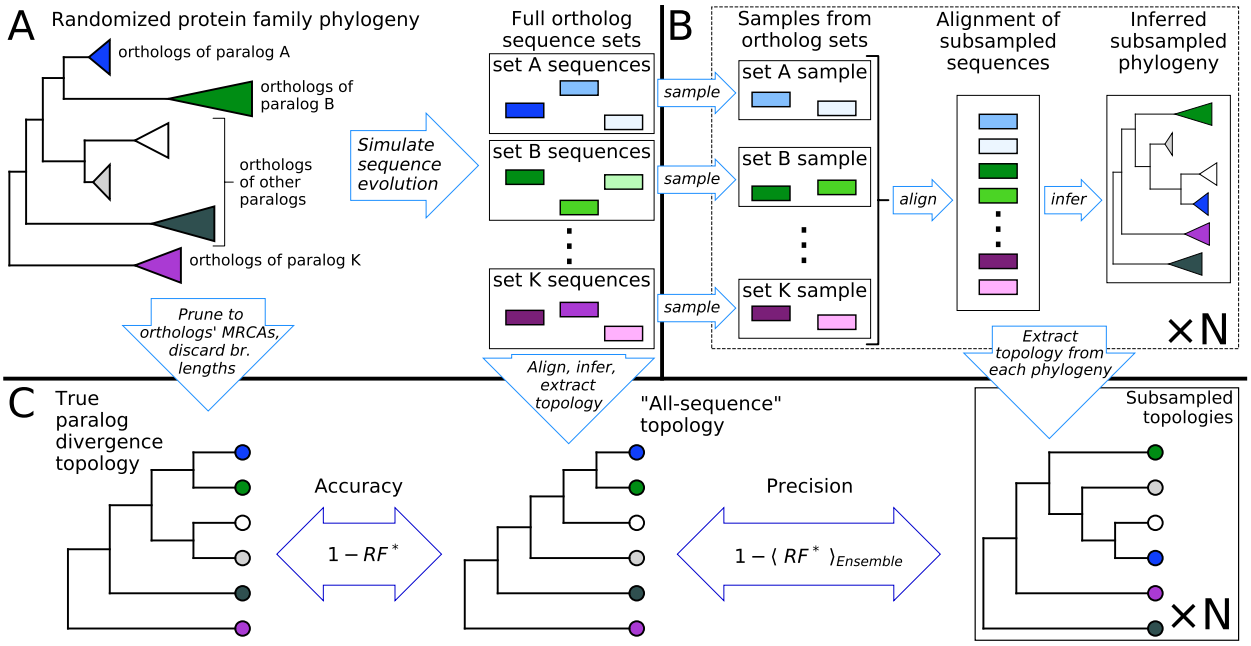

Figure 5.2.: Analysis framework for comparing reconstruction Accuracy and Precision

(**A**) Sequence evolution was simulated over synthetic phylogenies. Synthetic phylogenies were pruned to MRCAs of ortholog sets and branch lengths were discarded to obtain true paralog divergence topologies. Simulated sequences were aligned, phylogenies were inferred from those alignments, and "all-sequence" reconstructions of paralog divergence topologies were extracted. (**B**) Sequences were repeatedly sampled from each ortholog set in a family and phylogeny inference and topology extraction were done to produce a "subsampled topology". Repeating this N times yields an ensemble of topologies. (**C**) We define Accuracy as the similarity between the all-sequence reconstruction and Precision as the comparison between subsampled topologies and the all-sequence topology.

accuracy compared to the "true" divergence topology. Second, it allowed us to explore a range of divergence conditions by systematically varying branch lengths of input phylogenies, while controlling for other factors such as overall sequence length and the distribution of secondary structure elements and disordered loops. Assembling a comparable biological data set would have been impossible.

We simulated families containing 15 paralogs, each represented by 66 orthologs. In order to make the assessment statistically robust, we generated 600 families across a range

of post-duplication branch lengths. An alignment of human tyrosine kinase domains (median length 269 a.a.) was used as template for all simulations (see *Methods* for simulation details). We then used all combinations of three multiple sequence alignment algorithms (MAFFT's L-INS-i protocol[94], ClustalOmega[256], and Muscle[257]) and two phylogeny inference algorithms (FastTree2[258] and RAxML[259]) to reconstruct phylogenies for the 600 simulated families. We compared the reconstructed paralog divergence topologies, excluding speciation nodes by pruning orthologs' MRCAs to leaves, to the true divergence topology over which evolution was simulated (Figure 5.2A). We quantified topology differences with the Robinson-Foulds symmetric distance metric[260], modified to account for the occasionally non-monophyletic reconstruction of ortholog sets (RF^*, *Methods*). For convenience we define the accuracy of a reconstruction as $1 - RF^*$ distance between reconstructed and true paralog divergence topologies. Consistent with earlier studies[62,74,158–160,253,261], choice of alignment algorithm substantially affected accuracy, with L-INS-i alignments producing most accurate reconstructions, while FastTree2 and RAxML performed very similarly across all alignments (Supplementary Figure). Based on these results, we selected the combination of L-INS-i and FastTree2 for all remaining analysis.

5.3.1 Subsampling reveals an observable measure of accuracy

Given the known sensitivity of reconstruction to input alignment, we explored reconstruction variance resulting from differences in ortholog representation of paralogs using the framework outlined in Figure 5.2. We gathered the sets of ortholog sequences represent-

ing each paralog in a simulated family (Fig. 5.2A) and performed a resampling experiment (Fig. 5.2B): 50 times we randomly sampled 60 of 66 sequences (91%) from each ortholog set and performed traditional reconstruction, using L-INS-i and FastTree2 with each collection of subsampled input sequences. We retained a large fraction of sequences to minimize both the input variation and the loss of phylogenetic signal. To quantify reconstruction uncertainty, we measured the similarity (1− the average of RF^*) between topologies reconstructed from most of the sequences to the "all-sequence" topology (Fig. 5.2C). Since this quantity is a measure of how close the estimates are to each other, we refer to it as Precision.

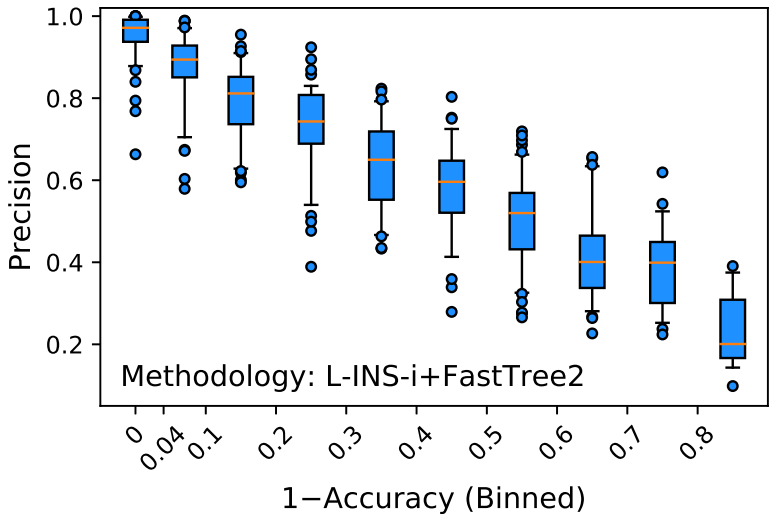

Figure 5.3.: Precision vs Accuracy of reconstruction

Reconstruction Precision plotted vs 1−Accuracy of all-sequence reconstruction for each simulated protein family. 1−Accuracy used on x-axis to make families with most accurate reconstructions appear on the left and those with least accurate on the right. Families were binned by 1−Accuracy. Tick marks on x-axis indicate bin boundaries.

Figure 5.3 demonstrates the striking relationship between accuracy of the all-sequence reconstruction (Accuracy) and Precision of reconstruction for families across a range of post-duplication branch lengths. Due to their strong correlation we use Precision, an observable

quantity for natural protein families, as a measure of a family's reconstruction Accuracy (unknowable for natural proteins) and, by proxy, the overall "complexity" of reconstruction for that family. Importantly, this also suggests that our 600 synthetic protein families span a range of complexities, allowing us to observe the performance of reconstruction as a function of complexity, via its proxy – Precision.

5.3.2 Using variability to distinguish phylogenetic signal from noise

Although we observed high reconstruction Precision for many families, only four of 600 families had identical paralog divergence topologies reconstructed from every subsampled alignment (Precision=1). Even among families with the highest Precision, and under dense subsampling, reconstruction variability was pervasive. On the other hand, Salichos and Rokas[255] argued that pairwise RF distances smaller than 1 (the average RF distance among randomly generated topologies) indicates consistent phylogenetic signal among the topologies being compared. Most of our 600 families had Precision ($1-\langle RF^* \rangle$) significantly greater than 0, but less than 1. Thus we sought to go a step further and test our central hypothesis: not only does intermediate Precision indicate consistent signal, but more frequently recapitulated features are more likely to be accurate, and this fact can be used to reconstruct more accurate topologies. In order to test this we first needed a way to extract frequently recapitulated features, and then a way to identify topologies most consistent with those features. Next we describe our method, ASPEN, which accomplishes both tasks.

5.4 Reconstructing topologies from ensemble sampling

We created a method we call ASPEN, for Accuracy through Subsampling of Protein EvolutioN, to construct and score topologies according to their consistency with topological features frequently represented in an ensemble of subsampled reconstructions (Fig. 5.2B). It relies on two key innovations: 1) extraction of topological features from an ensemble as frequencies of path lengths between leaves, and 2) an algorithm to construct and score topologies according to their consistency with observed path length frequencies.

5.4.1 Transforming topology sets into path length distributions

ASPEN's foundation is the equivalent representation of a topology (an acyclic, bifurcating graph) as a matrix of path lengths between leaves in terms of the number of internal nodes encountered along a path. First we demonstrate equivalence of graph and matrix representations by presenting a simple algorithm for interconverting between the two (Figure 5.4). Then we discuss how ensembles of topologies are transformed into path length frequency distributions.

Transforming a topology graph into a path length matrix

A topology can be equivalently represented as a matrix of leaf-to-leaf path lengths in terms of internal nodes encountered along the path. Transformation of a topology into its path lengths matrix representation is trivially accomplished by counting internal nodes along each path between pairs of leaves (Figure 5.4A).

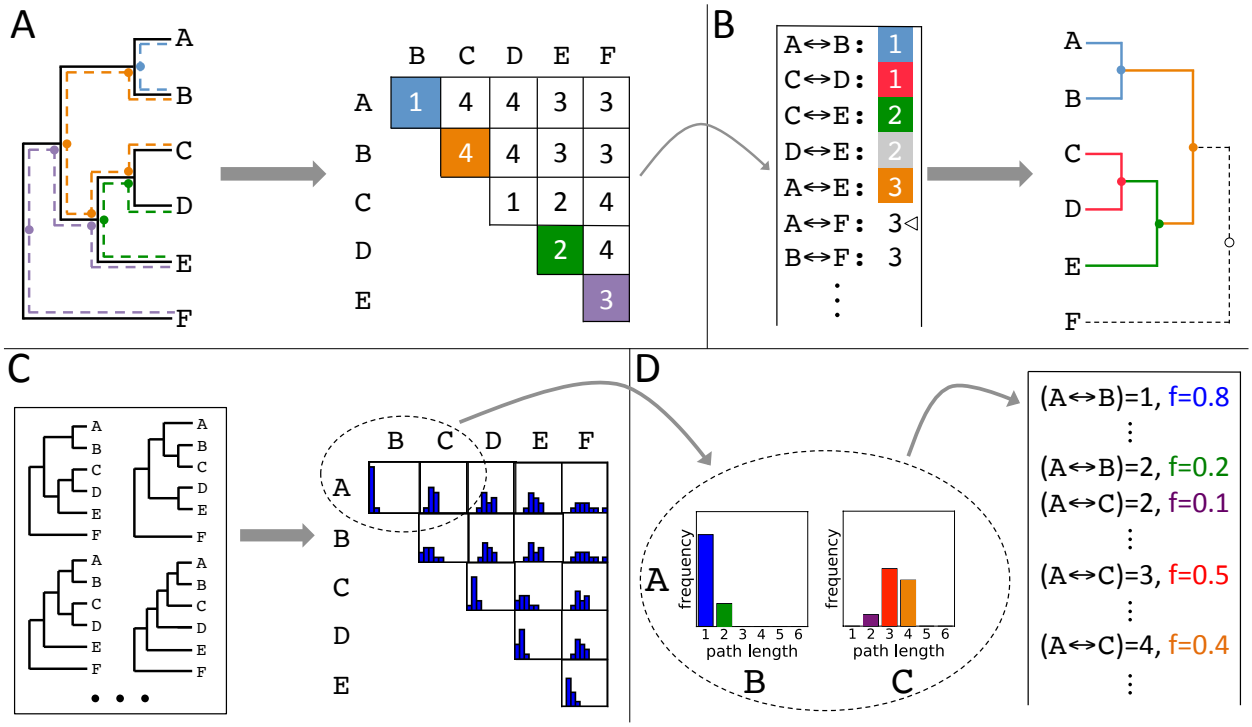

Figure 5.4.: Aggregating topological features across an ensemble of topologies using the path lengths matrix representation

(**A**) Decomposition of a topology into a matrix of leaf-to-leaf path lengths. Sample paths (A↔B,1), blue, (D↔E,2), green, (E↔F,3), violet, and (B↔C,4), orange, are highlighted. Dots indicate internal nodes along path. (**B**) Construction of a topology from a matrix of path lengths. First, the matrix is transformed into a sorted list of path lengths. Construction of internal nodes is triggered by path lengths encountered traversing the list: 1) Node $\{A, B\}$ joins leaves A and B and completes path $(A \leftrightarrow B, 1)$, blue. 2) Node $\{C, D\}$ joins leaves C and D and completes path $(C \leftrightarrow D, 1)$, pink. 3) Node $\{\{C, D\}, E\}$ joins leaf E to internal node $\{C, D\}$ and completes path $(C \leftrightarrow E, 2)$, green. Path $(D \leftrightarrow E, 2)$, grey, is completed by the same node and can be skipped during list traversal. 4) Node $\{\{A, B\}, \{\{C, D\}, E\}\}$ joins internal nodes $\{A, B\}$ and $\{\{C, D\}, E\}$ and completes path $(A \leftrightarrow E, 3)$, orange. Four paths of length 4 which appear further down the in the list are also completed by this node. Finally, 5) node $\{\{\{A, B\}, \{\{C, D\}, E\}\}, F\}$ joins leaf F to internal node $\{\{A, B\}, \{\{C, D\}, E\}\}$ and completes path $(A \leftrightarrow F, 3)$, dashed line. This completes the reconstruction, since all leaves are connected by the resulting topology. Path $(B \leftrightarrow F, 3)$ and all subsequent paths are already completed and can be ignored. (**C**) Each topology in the ensemble is decomposed into a matrix of leaf-to-leaf path lengths. Observed path lengths for each pair of leaves are aggregated into distributions. (**D**) Each distribution is then converted into a set of constraints on the length of the path between that pair of leaves. In the expanded section of the path lengths matrix, distributions of lengths for paths $(A \leftrightarrow B)$ and $(A \leftrightarrow C)$ are turned into constraints on the lengths of these paths by inserting each observed distance for each path, together with the frequency with which that distance was observed, into a list of path lengths. Vertical ellipses represent other paths of lengths 1, 2, 3, 4, etc. coming from elsewhere in the matrix.

Transforming a path length matrix into a topology graph

The reverse transformation can be accomplished using a simple bottom-up construction procedure (Figure 5.4B). Internal nodes are constructed by joining pairs of leaves and/or previously constructed internal nodes to recapitulate observed leaf-to-leaf path lengths. This bottom-up construction ("outside-in" for unrooted topologies) continues until all leaf nodes are connected by a single graph. Note that it is possible to encounter path lengths during list traversal which, at that state of construction, cannot be accommodated by constructing an internal node. For example, if the order of paths $(A \leftrightarrow E, 3)$ and $(A \leftrightarrow F, 3)$ in the list in Figure 5.4B were reversed and path $(A \leftrightarrow F, 3)$ was encountered first, it could not be accommodated because internal node $\{\{A, B\}, \{\{C, D\}, E\}\}$ would not yet be available to join to leaf F. Such path lengths are skipped and then revisited on the subsequent traversal of the list. Traversal is repeated as necessary until construction is completed. Because all path lengths are derived from a single topology, they are guaranteed to be consistent, making the construction unambiguous.

Generating path length frequency distributions

We take advantage of the alternate matrix representation to capture the individual variation of each leaf-to-leaf path length across an ensemble of topologies. Each topology is transformed into a path lengths matrix. Then path lengths for each pair of leaves are aggregated into a path length distribution for that pair (Figure 5.4C). Although ortholog sets overwhelmingly group into monophyletic subtrees across ensemble topologies (their MR-

CAs have no descendant leaves besides themselves), occasionally reconstructions do yield non-monophyletic ortholog sets. Because this violates an underlying assumption of the reconstruction, as well as the true topology of each synthetic protein family, we preclude paths compromised by this incorrect reconstruction from contributing to path length distributions: the length of any leaf-to-leaf path that contains a compromised internal node is not included in the distribution for that leaf pair.

5.4.2 Path length frequencies guide topology reconstruction

A score reflecting consistency with extracted features

ASPEN uses a quantitative metric for measuring the consistency of a proposed topology with observations from an ensemble of topologies. The score assigned to a topology is expressed in terms of log frequencies of leaf-to-leaf path lengths, $\log(f^L_{pair})$ where L is the length of path between leaves in *pair*, incorporated into the topology:

$$score = \sum_{\substack{leaf \\ pairs}} \log(f^L_{pair})$$

This scoring function rewards incorporation of frequently observed path lengths and penalizes rarely observed path lengths.

A branch-and-bound topology construction algorithm

Using the bottom-up procedure for constructing a topology graph from its path lengths matrix representation (Figure 5.4B), we developed an algorithm that uses a branch-and-bound strategy to construct the requested number of highest-scoring topologies according to the scoring function above. We describe the branching and bounding procedures in the next two sections.

Branching By analogy with the single-topology procedure in Figure 5.4B, construction of internal nodes is triggered by path length entries encountered during list traversal. However, this list contains every observed path length for every leaf pair, together with its frequency (Figure 5.4D). Unlike the single-topology case, list entries cannot be assumed to be consistent with each other. In fact, many combinations of path lengths on the list cannot be incorporated into one topology. For example, for hypothetical leaves A, B, and C, path lengths $(A \leftrightarrow B, 1)$ and $(B \leftrightarrow C, 1)$ are mutually exclusive because in a bifurcating topology B can be one internal node removed from either A or C, but not both. In single topology reconstruction, if a path length could be completed by the introduction of an internal node, that node could be safely constructed because it was guaranteed to satisfy every other list entry. Since that guarantee no longer holds, multiple topologies are constructed simultaneously by allowing the construction path to branch (Figure 5.5).

"Assemblies" are used to track simultaneous reconstruction of multiple topologies. Each assembly holds a copy of the path length frequencies list, a partially constructed topology, and the current topology score according to the scoring function (discussed below in the

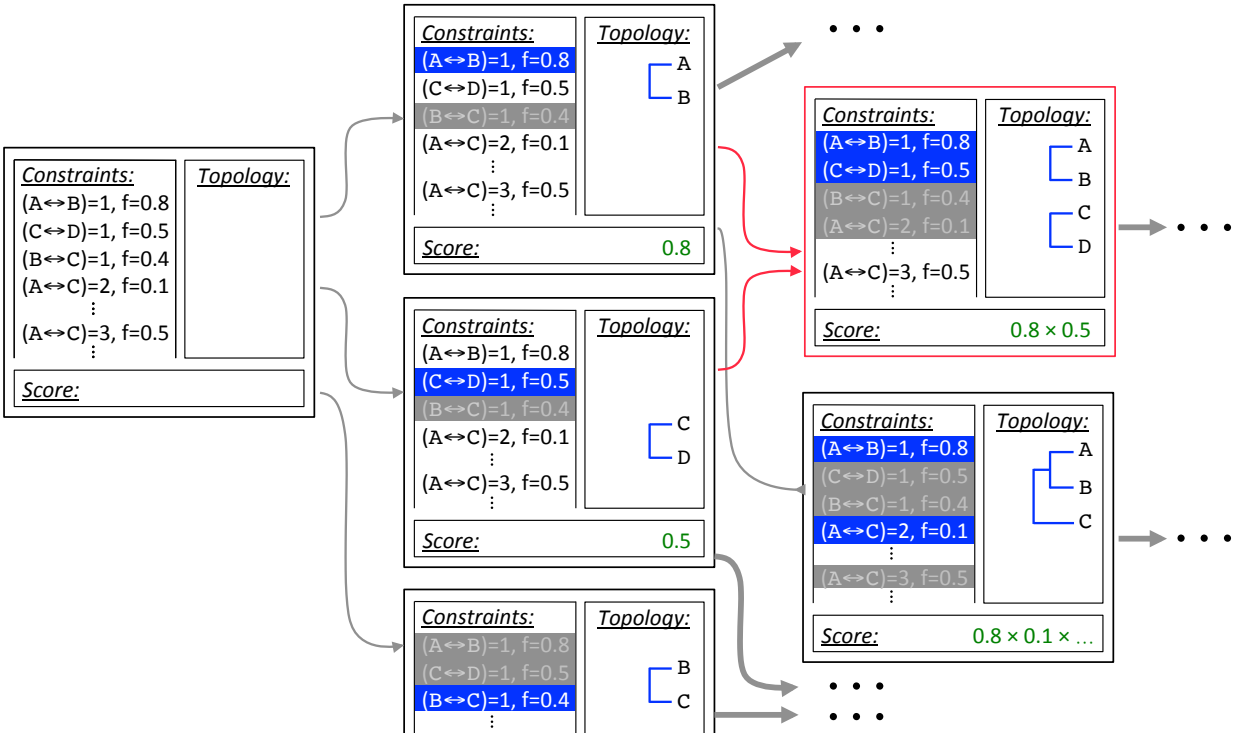

Figure 5.5.: Branching construction of topologies by incorporating path lengths observed in an ensemble

Construction begins with the empty topology assembly on the left. Every possible extension is constructed in a copy of the initial assembly: Node $\{A, B\}$ completes path $(A \leftrightarrow B, 1)$, node $\{C, D\}$ completes path $(C \leftrightarrow D, 1)$, and node $\{B, C\}$ completes path $(B \leftrightarrow C, 1)$, branching the initial assembly into three new assemblies. Path lengths completed by the introduced node and path lengths incompatible with it are marked and not revisited. Nodes $\{A, B\}$ and $\{C, D\}$ preclude path $(B \leftrightarrow C, 1)$, while node $\{B, C\}$ precludes paths $(A \leftrightarrow B, 1)$ and $(C \leftrightarrow D, 1)$. Completed paths are shown in blue, precluded paths are greyed out in the corresponding assemblies. Intermediate topology scores are calculated according to the scoring function. On the next iteration construction paths for assemblies $\{A, B\}$ and $\{C, D\}$ collide, indicated in red. A single copy of the resulting assembly, $\{A, B\}, \{C, D\}$, is retained. Assembly $\{A, B\}$ is separately extended with node $\{\{A, B\}, C\}$. Additional construction paths, indicated by ellipses, are not shown.

section on bounding). Reconstruction proceeds in iterations, starting with a single empty assembly (Figure 5.5, left). On the first iteration, the entire list is traversed and *every* possible extension by introduction of a new node is created simultaneously in a copy of the original assembly (Figure 5.5, middle). In each new assembly, all path lengths completed

by the new node and all path lengths incompatible with it are marked and not re-examined on subsequent iterations. Remaining path lengths are not completed by the new node, but remain compatible with it. On subsequent iterations the same procedure is repeated for all tracked assemblies.

In principle, branching and iteration alone yield every topology consistent with path lengths observed in the ensemble. In practice, this results in a combinatorial explosion which must be carefully managed to allow construction to proceed to completion. First, Figure 5.5 (right) demonstrates how branching to satisfy non-conflicting path lengths can lead to collisions between diverged construction paths on later iterations. This occurs because many topologies can be constructed by introducing internal nodes in multiple orders. Each branched path represents a particular order of internal node introduction. In a practical implementation collisions must be managed in order to prevent multiple reconstructions of the same topology by multiple paths – an enormous replication of effort.

Second, even if each distinct topology is constructed once, in most cases reconstructing every topology consistent with observations from the ensemble, no matter how infrequent, is neither practical nor useful. Bounding, described in the next section, guarantees reconstruction of only the requested number of top scoring topologies.

Bounding The score is used to rank completed topologies, where ranking is updated every time a new topology is finished. The number of top scoring topologies to reconstruct, X, is requested at the beginning of a reconstruction run (10,000 was used in ASPEN evaluation). Once the initial X topologies are constructed, the Xth topology score constitutes the bound.

Partially constructed topologies are abandoned if no complete topology that can be derived from that construction state will score above the bound. We determine this by calculating the score for already-incorporated path lengths and projecting the best possible score for a complete topology by assuming the most frequent remaining path length will be incorporated for every unconnected leaf pair:

$$projected = \sum_{\substack{incorporated \\ paths}} \log(f^L_{path}) \quad + \sum_{\substack{remaining \\ paths}} \max(\log(f^L_{path}))$$

As more high-scoring topologies are constructed, the bounding criterion becomes more strict allowing both more and earlier abandonment of unproductive construction paths. The branch-and-bound strategy guarantees that the X topologies remaining on the list at the end of a run are the X highest scoring topologies according to the scoring function.

5.5 Evaluation and Discussion of ASPEN reconstructions

To test our algorithm, for each protein family we generated ensembles of 1000 subsampled topologies with each ortholog set represented by 30 of 66 orthologs (≈45%). Then we used ASPEN to reconstruct 10,000 top scoring topologies for two-thirds of the families. Because accuracies of all reconstructions vary substantially across the range of reconstruction Precision, as does the relative accuracy of ASPEN-reconstructed topologies, the families were binned by their Precision for the purposes of this analysis. Next we examine the relationship between reconstruction Precision and the discriminatory power of the log-frequency func-

tion with respect to accuracy, and then compare ASPEN reconstructions with all-sequence reconstructions and discuss the implications of our observations.

5.5.1 Log-frequency score is correlated with accuracy

To understand the relationship between the log-frequency score and the accuracy of reconstructed topologies, we plotted the ASPEN topology rank vs the bin-average accuracy of topologies (Figure 5.6B-G). Among higher-Precision families (Figure 5.6B-D), top-ranked log-frequency scores are strongly correlated with accuracy, particularly for topologies ranked in the top $\sim$50, which indicates the independent scoring function based on observed frequencies across the ensemble are indicative of accuracy. The strength of correlation decreases as difficulty of reconstruction increases (lower Precision bins, Figure 5.6E-G), indicating less discriminatory power with respect to accuracy. Nevertheless, ASPEN's top-ranked topology is, on average, also its most accurate across all Precision bins.

5.5.2 Top ASPEN topology beats all-sequence reconstructions

Next, we compared ASPEN's best topology to all-sequence single-alignment reconstructions (Figure 5.6A). Like all other methods, ASPEN's accuracy is a function of Precision, or difficulty of the reconstruction task. As discussed earlier, MAFFT L-INS-i alignments yielded the most accurate all-sequence reconstructions across all Precision bins, while FastTree2 and RAxML performed very similarly on all alignments. Both top-ranked ASPEN topologies and L-INS-i all-sequence reconstructions have nearly perfect accuracy on families in the highest-

Precision bin – not particularly surprising, since subsampled topology ensembles for ASPEN reconstruction were generated using the combination of L-INS-i and FastTree2 (*Methods*). Much more intriguing is the fact that top-ranked ASPEN topologies are consistently more accurate than any all-sequence reconstruction across the remaining Precision bins. Moreover, although the accuracy of all reconstructions degrades with difficulty of the reconstruction task (lower Precision), ASPEN's accuracy degrades much more slowly. ASPEN's top topology provides the greatest accuracy improvement over single-topology reconstructions when reconstruction is the most difficult.

5.5.3 ASPEN produces many accurate topologies at low Precision

To compare more ASPEN topologies with the most accurate all-sequence reconstructions, bin-average accuracies of L-INS-i / FastTree2 topologies are plotted alongside bin-average accuracies of top-300 ranked topologies (Figure 5.6B-G). Although the log-frequency score provides less discrimination with respect to accuracy, more of ASPEN's topologies outperform single-alignment reconstructions as Precision decreases and reconstruction becomes harder. In the two lowest-Precision bins (Figure 5.6E-G), all top-300 ASPEN topologies are more accurate than the best all-sequence reconstruction. Taken together, these observations suggest that ASPEN results should be considered differently for families with high and low reconstruction Precision.

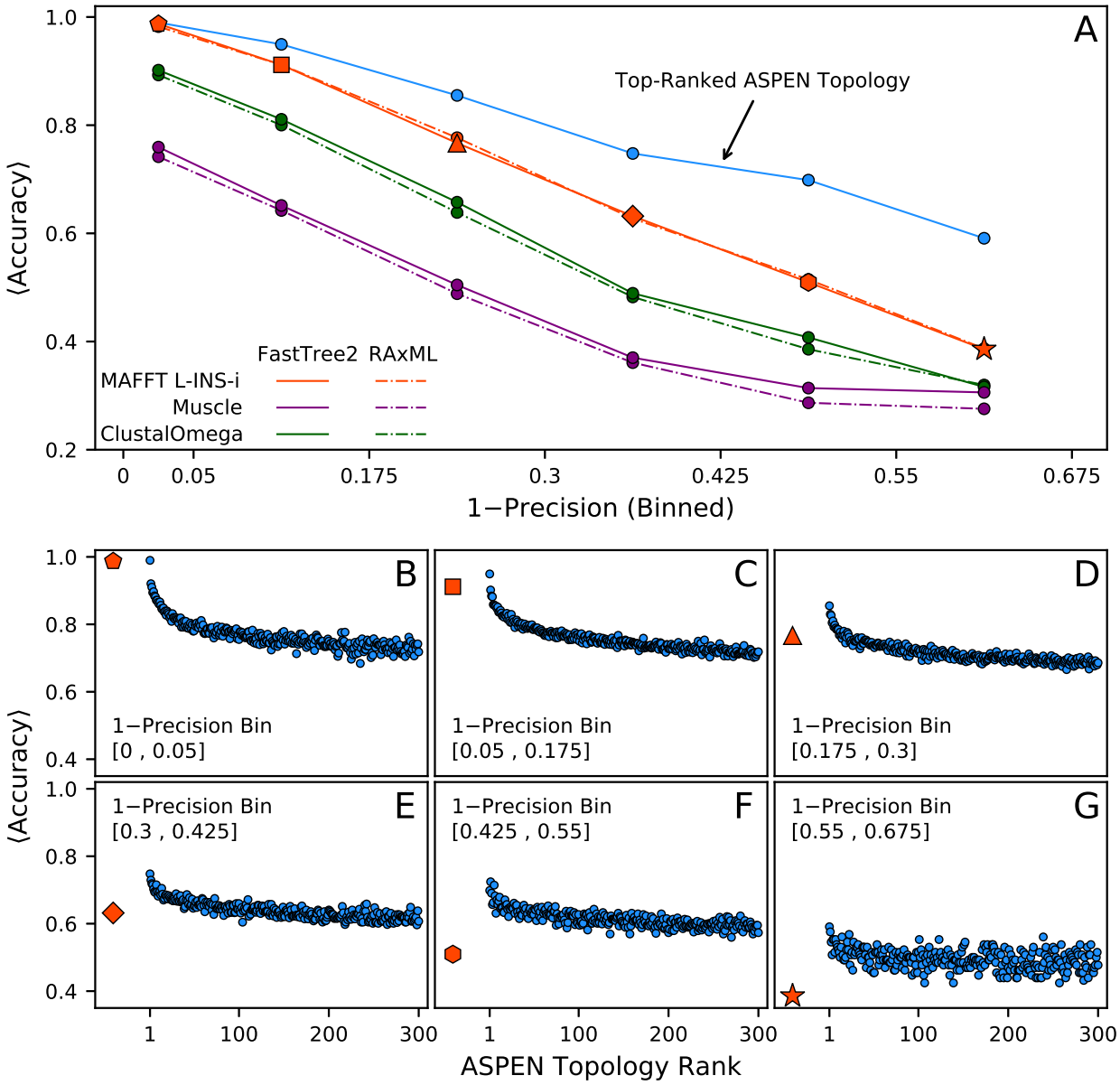

Figure 5.6.: Accuracy of topologies reconstructed by ASPEN

(**A**) Accuracy, as a function of 1−Precision of a family's reconstruction, of the top-ranked ASPEN topology and all-sequence reconstructions. Families were binned according 1−Precision. Ticks on x-axis correspond to bin edges. Average accuracy of each type of reconstruction across families in the bin is plotted. For all-sequence reconstructions with MAFFT L-INS-i and FastTree2 (orange, solid line) a unique marker shape is used in each Precision bin. (**B**)-(**G**) For each Precision bin in (**A**), accuracy of ASPEN topologies ranked 1 through 300, averaged for each rank across all families in the bin, plotted as a function of rank. Average accuracy of the L-INS-i / FastTree2 all-sequence reconstruction is plotted for comparison on the left of each panel.

5.5.4 How to use ASPEN in different Precision conditions

The top few topologies are best for high-Precision families

For families with high Precision, where one may reasonably expect to reconstruct an accurate topology, ASPEN's top, or top few topologies are likely more accurate than any single-alignment reconstruction. One or a few of these topologies can be confidently used for downstream applications. This result is far from trivial, given that ASPEN's subsampling approach scales far better with the overall number of input sequences than traditional statistical reconstruction methods. With the advent of affordable genome sequencing and the resulting explosion in the number of sequenced and annotated species' genomes[262,263], all-sequence reconstruction of paralog divergence by statistical methods has become infeasible for many families with large numbers of orthologs. Therefore, subsampling large samples of orthologs to yield a Precision score can now be used to identify how likely the full sequence topology is to be accurate, determining if one is working in a high or low Precision/Accuracy regime.

Diverse representation is critical at lower Precision

Accuracy of all reconstructions suffers for families with lower reconstruction Precision (greater difficulty for reconstruction). Even top-ranked ASPEN topologies cannot be expected to be completely accurate. In this Precision regime all of the top 300 ASPEN topologies, or more, can be considered comparably plausible models, given the sequence data.

Since all of these topologies are very likely to be more accurate than any individual single-alignment reconstruction, under these conditions ASPEN topology reconstruction should be treated as a mechanism for sampling a large number of imperfect, but quite accurate topologies. As the true topology cannot be distinguished from other, fairly accurate topologies on the basis of such sequence data, any downstream analysis relying on a divergence topology should aim to integrate over this topological uncertainty.

5.6 Conclusion

Subsampling in the process of reconstruction proved to be extremely powerful – it identified two measures (Precision and Score based on observed frequencies) of something unknowable (Accuracy) and guided a reconstruction method that identifies much more accurate topologies than traditional approaches. That ASPEN reconstructions were more accurate than single-alignment reconstructions, is evidence that the central hypothesis of this work is supported – relationships found consistently amongst the variance produced by subsampling are more likely to be reflective of true protein divergence histories. We anticipate that, as a meta analysis approach to tree evaluation and reconstruction, ASPEN is likely to continue to boost the accuracy of individual approaches.

We also conclude from this study that it is worth revisiting the reconstruction accuracy of real protein families, particularly for those widely relied-upon reconstructions[29,30,252]. The reconstruction of proteins from a single alignment of small numbers of orthologs may suffer from the same or worse accuracy issues we saw in single alignment approaches of our synthetic

family. They may be worse in accuracy than what we observed in this study, since such reconstructions are derived from much smaller subsamples of ortholog sequences than we used in our subsample presented here and we found for small subsamples even for relatively high-Precision families individual reconstructions are extremely unreliable.

5.7 Materials and Methods

5.7.1 Preparation of synthetic sequence data

All sequence simulation materials and simulated sequence alignments are available via Figshare (10.6084/m9.figshare.5263885).

Construction of phylogenies representing protein family divergence

Random 15-leaf phylogenies representing paralog divergence were generated at http://www.trex.uqam.ca[264] using the procedure of Kuhner and Felsenstein[265]. 100 phylogenies were generated with each average branch length of 0.5, 0.6, 0.7, 0.8, 0.9, and 1.0, 600 in all. The Ensembl Compara species tree topology[266] containing 66 metazoan species was used for the divergence of each ortholog set. The topology was parametrized with branch lengths corresponding to species divergence times at http://www.timetree.org[267,268]. For each of 15 leaves in each random phylogeny, a copy of the parametrized species tree was randomly scaled in overall height and had each individual branch randomly perturbed around its true length to maintain a realistic scale of divergence. The roots of these ran-

domized trees (representing the MRCAs of an ortholog sets) were grafted at the leaves of the paralog phylogenies, resulting in 990-leaf synthetic protein family phylogenies.

Preparation of sequence template and sequence simulation

Human tyrosine kinase domains were aligned using MAFFT L-INS-i with default parameters. This alignment was used as the template for sequence simulations as follows. The alignment was divided into 24 segments on the basis of local sequence similarity and analysis of solved tyrosine kinase structures. Each segment was assigned a substitution rate scaling factor and an insertion/deletion model to match degree of conservation and solvent exposure in solved structures. Simulation was carried out over synthetic phylogenies using indel-seq-gen[269–271] under the JTT substitution model.

5.7.2 Phylogeny reconstruction

All-sequence phylogenies were inferred using all combinations of MAFFT L-INS-i, ClustalOmega, and Muscle for sequence alignment and of FastTree2 and RAxML for phylogeny inference. Subsampled phylogenies for Precision calculations (60 of 66 orthologs sampled from each ortholog set, 50 phylogenies reconstructed per protein family) were inferred with FastTree2 only, due to run time considerations. Subsampled phylogenies for ensembles (30 of 66 orthologs sampled, 1000 phylogenies per protein family) were reconstructed using L-INS-i and FastTree2 only.

Alignment algorithms were used with their default settings. FastTree2 was used with default settings and the WAG substitution model. RAxML was used with default settings and the PROTGAMMAWAGF variant of the WAG substitution model. The WAG substitution model was deliberately used for topology inference, instead of the JTT substitution model used for simulating protein families, in order to emulate the more realistic scenario where models used for reconstruction of phylogenies for natural families do not precisely match the substitution patterns in those families.

Accuracy and Precision of reconstruction for a protein family are defined in terms of the L-INS-i / FastTree2 all-sequence and subsampled topology reconstructions.

5.7.3 Modified Robinson-Foulds topology comparison metric

The Robinson-Foulds (RF) metric is defined in terms of leaf partitions at internal topology nodes for two topologies with identical sets of leaves. For a tree with N leaves there are $N - 3$ informative splits. The normalized form of the Robinson-Foulds comparison metric for two topologies, A and B, is:

$$RF = \frac{x + y}{2N - 6} \tag{5.1}$$

Where x is the number of leaf partitions in A but not in B, y is the number of leaf partitions in B but not in A, N is the number of leaves in each topology, and $2N - 6 = 2 \times (N - 3)$ is the number of informative splits in the two topologies.

In order to compare reconstructed paralog divergence topologies we had to modify the RF metric to accommodate cases when the MRCA of an ortholog set has as descendants one or more MRCAs of other ortholog sets. Such topologies are poorly formed because they require inference of additional unobservable events – loss of paralogs in some lineages – in order to be reconciled with a duplication/speciation divergence history. Because the offending ortholog set cannot be pruned to a leaf MRCA, the resulting topology cannot be compared to properly formed topologies (e.g. the true topology) using the standard RF metric. In effect, when ortholog leaves and speciation internal nodes of the offending ortholog set are pruned, the resulting topology is missing a MRCA leaf, because that MRCA maps to an internal node, making that node ambiguous in its duplication vs speciation status. This is problematic for RF because it affects the denominator. Nevertheless, their internal nodes representing pre-duplication common ancestors of the offending ortholog set/paralog and other paralogs can match, in terms of induced partition of *paralogs*, equivalent nodes in other topologies.

In the modified RF^*, N represents the number of paralogs (ortholog sets) in each compared topology, not the number of leaves. In addition to x and y we define z as the number of MRCA leaves missing from A but not from B and z' as the number of MRCA leaves missing from B but not from A. The modified metric is then calculated as:

$$RF^* = \frac{x + y + z + z'}{2N - 6} \tag{5.2}$$

5.7.4 ASPEN

ASPEN is implemented in python 2.7. The ASPEN development repository is publicly available at *https://github.com/NaegleLab/ASPEN*.

5.8 Supplementary Materials

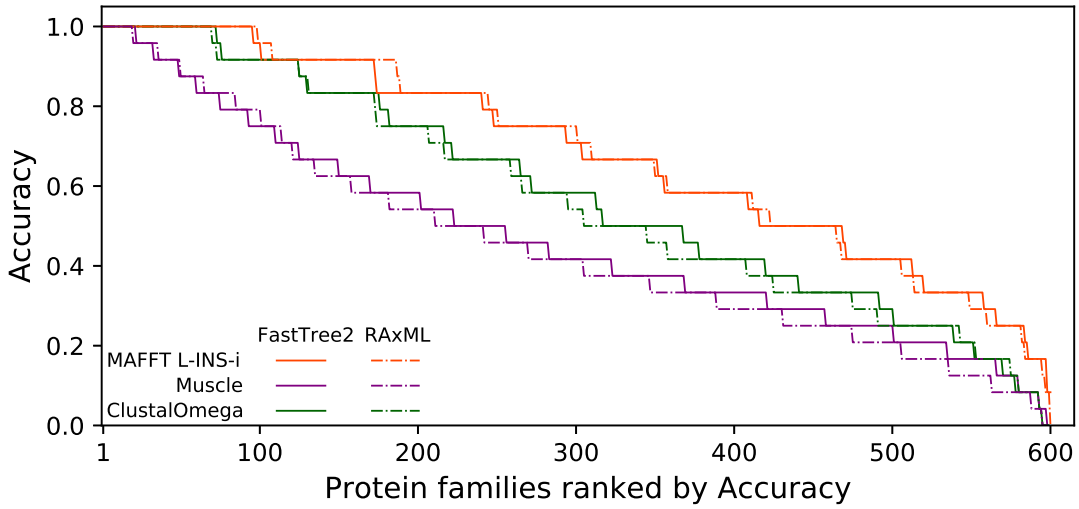

Figure 5.7.: Supplementary Figure

Accuracy of all-sequence reconstructions by all combinations of alignment and phylogeny inference tools plotted against protein family rank according to same reconstruction accuracy.

5.9 Acknowledgments

This work was enabled by the Center for High Performance Computing in the Mallinkrodt Institute of Radiology at Washington University in St. Louis and the Center for Biological Systems Engineering. We wish to thank Tom Ronan, Dr. Barak Cohen, Dr. Gary Stormo, Dr. Justin Fay, and Dr. Jim Havranek for the helpful discussions that shaped this work.

6. Conclusions and Future Directions

6.1 Why analysis algorithms can fail to identify structure in data

The overarching theme of work presented in this dissertation was identifying cases when existing, somewhat naive analysis algorithms fail to detect the complex hidden structure in biological data, and then adapting and extending those algorithms to handle the complexity. In some cases, actual structure is convoluted with and obscured by noise, leading to interpretation of noise artifacts as spurious structure. This is often evidenced by conflicting analyses resulting from multiple, equally valid applications of an algorithm. For example, numerical data structured in poorly separable clusters may yield different partitions on different runs of a randomly initialized clustering algorithm. In other cases, actual structure cannot be detected because it violates underlying assumptions explicitly or implicitly encoded in existing algorithms. An example of this is the implicit assumption by clustering algorithms of globular cluster shape rendering clusters with more complex shapes undetectable[17]. Data resampling is a powerful tool for detecting the unreliability of individual solutions and for separating actual structure from random noise. Unfortunately, I know of no systematic way of identifying cases when data violates underlying assumptions of analysis methods.

In this work I described two cases – clustering noisy numerical data and reconstructing paralog divergence topologies – where resampling approaches led to substantial improvements

in accuracy. However, in a third case – detection of partial SDPs – resampling only led to a marginal improvement. A much more substantial improvement resulted from modifying an SDP detection algorithm in light of the key insight that partially conserved positions may have strong SDP signal. Formalizing the investigative approach which led to this insight is beyond the scope of this thesis.

6.2 Robust data partitioning with parametric and non-parametric resampling

6.2.1 Ensemble clustering with non-parametric sampling

In chapter 2 I described applying ensemble clustering with sampling over clustering algorithms, data transformations, distance metrics, and cluster number, using the approach and software originally described by Naegle et al.[211] to analyze the role of nuclear export of HDAC5 in injury response. We found that export of HDAC5 from the nucleus activates a pro-regenerative transcriptional program, including export-dependent up-regulation of transcription factors *jun*, KLF4 and KLF5, *Fos* and *Gadd45a*.

In chapter 3 I described permutation sampling from replicate time course data as a means of bootstrapping a non-parametric model for the noise present in a high-throughput phosphoproteomic data set. We built an ensemble of clustering solutions over multiple data sets resampled under this noise model and identified robustly co-clustering phosphopeptide time courses.

6.2.2 Ensemble clustering with sampling from parametric noise models

Another approach I described in chapter 3 is using parametric distributions to model noise. We built an ensemble of clustering solutions from data sets constructed by perturbing measurements with noise sampled from a parametric noise model. We selected for this analysis a "foreground" gene expression data set that contained no replicates and two "background" data sets with replicates collected using the same microarray platform. We used the background data to model the mean-dependent variance for the foreground data set. Background data sets were rescaled to the 75th quantile of the foreground dataset, which produced a high degree of agreement in probe intensity across the entire range of all data sets, indicating that background data could indeed be used to model noise for the foreground. Finally, we used the program Cyber-T[216] to generate a mapping from mean $\log_2$(expression) to standard deviation, the second parameter needed to parametrize a noise distribution.

6.3 Detection of partially conserved Specificity Determining Positions

The LacI family had previously been used to test several algorithms for detecting Specificity Determining Positions[126,129,132,139,144]. Only a small fraction of available LacI sequences (52 of thousands) representing a subset of known LacI paralogs (15 out of at least 20 represented by 28 or more sequences) were used in all of those analyses because the same alignment originally produced in 2002 was re-analyzed by each method.

6.3.1 Nearly all SDPs are only used by a subset of LacI paralogs

In chapter 4 I described the SDP analysis I performed on a much larger LacI data set containing 1814 unique sequences representing 20 paralogs. I found that the group-wise conservation pattern associated with SDPs was present in as many as one third of all sequence positions, but that no more than 15, and usually closer to 10 paralogs, were group-wise conserved at any position. After implementing a modified scoring function capable of detecting partial group-wise conservation, I identified 10 to 20 positions with strong SDP signal in each paralog. Critically, each paralog's complement of SDPs was unique, suggesting that evolution uses a strategy of mixing and matching positions capable of contributing to specificity in order to determine the unique specificity of each paralog.

6.3.2 Additional group-wise conservation among higher order groups

In addition to the numerous paralog-wise conserved positions, I identified several positions with a group-wise conservation pattern, where subsets of paralogs were conserved to the same amino acid. When each ortholog set was treated as a specificity group, such positions did not have particularly strong SDP signal because of relatively high between-group agreement. However, these positions may in fact be specificity determinants shared by multiple paralogs. I found this higher-order conservation pattern to be much more prevalent among SH2 domains (unpublished data), which are both substantially more numerous than LacI paralogs (120 in the human genome) and may overlap in their recognition specificity in non-trivial ways[272,273]. Detecting specificity determinants among domain families with this

conservation pattern is likely to require a dynamic definition of specificity groups which can be modified from position to position.

6.4 Reconstructing divergence histories of protein families

The genesis of this project was my attempt to infer a phylogeny for 3000+ scrupulously curated SH2 domain sequences. Since I was only interested in the divergence of paralogs, not the subsequent divergence of orthologs, I abandoned the all-sequences approach after I got tired of waiting for the maximum likelihood reconstruction to finish. Instead, inspired by my earlier work resampling numerical data, I decided to try a subsampling strategy with ortholog sequences, naively expecting to a single paralog divergence topology to dominate all resampled reconstructions. Instead, to my great surprise and deep consternation, nearly every resampled reconstruction was unique. After replicating this result with numerous simulated protein families, I came to doubt the reliability of any single topology reconstruction for any protein family.

6.4.1 Subsampling to assess accuracy of single-alignment reconstructions

Simulating sequence evolution affords one the luxury of knowing the true phylogenetic tree according to which the sequences diverged. This allowed me to measure the accuracies of various reconstructions and to correlate accuracy with other quantities which can be measured from the reconstruction(s) alone. Unlike accuracy, such quantities are observable for natural, as well as simulated families. In chapter 5 I described the strong correlation I

discovered between the accuracy of a single-alignment, all-sequence reconstructions of paralog divergence and a quantity I call precision of reconstruction, defined in terms of agreement among subsampled reconstructions. This is a critical finding that provides the first means of assessing how accurate a traditional single-alignment reconstruction is likely to be.

6.4.2 Building topologies from common features of subsampled reconstructions

While paralog divergence reconstructions from subsampled sequence sets frequently, and alarmingly, disagree with each other, topological features on which the reconstructions *do* agree are most likely to reflect the true divergence topology. ASPEN, the topology reconstruction algorithm I described in chapter 5, takes advantage of this fact by extracting shared topological features from an ensemble of subsampled topology reconstructions in the form of frequently occurring path lengths between paralogs in terms of internal topology nodes representing pre-divergence common ancestors. ASPEN incorporates the frequencies with which specific path lengths between leaves were observed into its scoring function and uses a branch-and-bound strategy to build topologies containing, according to the scoring function, the most frequently observed topological features. Resulting topologies are more accurate than single-alignment reconstructions.

6.4.3 Improving ASPEN

ASPEN already provides a powerful means to reconstruct paralog divergence, but I would like to implement a number of improvements to its methodology. Although the branch-and-

bound guarantee that all highest-scoring topologies have been identified only holds if ASPEN reconstruction runs to completion, I frequently observe drastically diminishing returns as a reconstruction progresses. For the number of highest-scoring topologies requested at the beginning of a reconstruction, the very best topologies are usually identified early on, while the bottom of the requested ranked list takes much longer to finalize. Empirically estimating the likelihood of encountering a new top-scoring topology, given dynamics observed in the ranked list of topologies so far, would allow terminating a reconstruction run early with high confidence that all of the best topologies have been identified. Furthermore, a greater empirical understanding of the relationship between the frequency-based score and topology accuracy might facilitate rationally selecting the number of top-scoring topologies that should be used for downstream applications. Finally, the relationship between subsample size during assembly of the ensemble of subsampled topologies, the duration of the subsequent reconstruction, and the accuracy of resulting topologies remains to be explored. Anecdotally, subsample size can affect both reconstruction accuracy an run time, so it may be used to tune the desired balance between the two.

6.5 Downstream applications of paralog divergence reconstruction

I believe the findings about group-wise conservation patterns and paralog divergence reconstruction described in chapters 4 and 5 form a strong foundation for functional evolutionary analysis of large and complex protein or protein domain families. I am currently using ASPEN to reconstruct the divergence history of SH2 domains in order to facilitate

dynamic specificity group redefinition in identification of specificity determinants. Because many SH2 domains appear to pairwise overlap in their recognition specificity, I believe their residue-level specificity encoding may be deeply hierarchical, with some SDPs shared by large domain classes and others differentiating individual paralogs or small paralog groups. This approach may also be applied to dissect the residue-level logic which encodes the binding specificity of other recognition domains involved in signal transduction, such as SH3, PDZ, WW, PH, and, more broadly, to analyze the functional divergence of any protein family containing many paralogs. I believe this kind analysis can be used to generate targeted hypotheses which would allow dissecting protein function in a manageable number of experiments.

7. Bibliography

[1] Hilbert, M. & López, P. The world's technological capacity to store, communicate, and compute information. *Science* **332**, 60–5 (2011).

[2] Slonim, D. K. From patterns to pathways: gene expression data analysis comes of age. *Nature Genetics* **32 Suppl**, 502–8 (2002).

[3] Mortazavi, A., Williams, B. A., McCue, K., Schaeffer, L. & Wold, B. Mapping and quantifying mammalian transcriptomes by rna-seq. *Nature Methods* **5**, 621–8 (2008).

[4] Lister, R. *et al.* Human dna methylomes at base resolution show widespread epigenomic differences. *Nature* **462**, 315–22 (2009).

[5] Sabo, P. J. *et al.* Genome-scale mapping of dnase i sensitivity in vivo using tiling dna microarrays. *Nature Methods* **3**, 511–8 (2006).

[6] Hesselberth, J. R. *et al.* Global mapping of protein-dna interactions in vivo by digital genomic footprinting. *Nature Methods* **6**, 283–9 (2009).

[7] Huang, P. H. & White, F. M. Phosphoproteomics: unraveling the signaling web. *Molecular Cell* **31**, 777–81 (2008).

[8] Gajadhar, A. S. & White, F. M. System level dynamics of post-translational modifications. *Curr Opin Biotechnol* **28**, 83–7 (2014).

[9] Patti, G. J., Yanes, O. & Siuzdak, G. Innovation: Metabolomics: the apogee of the omics trilogy. *Nature Reviews. Molecular Cell Biology* **13**, 263–9 (2012).

[10] Jain, A. K., Murty, M. N. & Flynn, P. J. Data clustering: A review. *ACM Comput. Surv.* **31**, 264–323 (1999).

[11] Andreopoulos, B., An, A., Wang, X. & Schroeder, M. A roadmap of clustering algorithms: finding a match for a biomedical application. *Briefings in Bioinformatics* **10**, 297–314 (2009).

[12] Xu, R. & Wunsch, D. C. Clustering algorithms in biomedical research: A review. *IEEE Reviews in Biomedical Engineering* **3**, 120–154 (2010).

[13] Tavazoie, S., Hughes, J. D., Campbell, M. J., Cho, R. J. & Church, G. M. Systematic determination of genetic network architecture. *Nature Genetics* **22**, 281–285 (1999).

[14] Golub, T. R. *et al.* Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science (New York, N.Y.)* **286**, 531–7 (1999).

[15] Naegle, K. M., Welsch, R. E., Yaffe, M. B., White, F. M. & Lauffenburger, D. A. Mcam: multiple clustering analysis methodology for deriving hypotheses and insights from high-throughput proteomic datasets. *PLoS Comput Biol* **7**, e1002119 (2011).

[16] Naegle, K. M., White, F. M., Lauffenburger, D. A. & Yaffe, M. B. Robust co-regulation of tyrosine phosphorylation sites on proteins reveals novel protein interactions. *Mol Biosyst* **8**, 2771–82 (2012).

[17] Ronan, T., Qi, Z. & Naegle, K. M. Avoiding common pitfalls when clustering biological data. *Sci Signal* **9**, re6 (2016).

[18] Köppen, M. The curse of dimensionality. In *5th Online World Conference on Soft Computing in Industrial Applications (WSC5)*, 4–8 (2000).

[19] Parsons, L., Haque, E. & Liu, H. Subspace clustering for high dimensional data: A review. *SIGKDD Explor. Newsl.* **6**, 90–105 (2004).

[20] Fred, A. L. N. & Jain, A. K. Data Clustering Using Evidence Accumulation. In *Proceedings of the 16th International Conference on Pattern Recognition*, 276–280 (2002).

[21] Pavlidis, P., Li, Q. & Noble, W. S. The effect of replication on gene expression microarray experiments. *Bioinformatics* **19**, 1620–1627 (2003).

[22] Posekany, A., Felsenstein, K. & Sykacek, P. Biological assessment of robust noise models in microarray data analysis. *Bioinformatics (Oxford, England)* **27**, 807–14 (2011).

[23] Kerr, M. K. & Churchill, G. A. Bootstrapping cluster analysis: Assessing the reliability of conclusions from microarray experiments. *Proceedings of the National Academy of Sciences* **98**, 8961–8965 (2001).

[24] Patwardhan, R. P. *et al.* High-resolution analysis of dna regulatory elements by synthetic saturation mutagenesis. *Nature Biotechnology* **27**, 1173–1175 (2009).

[25] Druley, T. E. *et al.* Quantification of rare allelic variants from pooled genomic dna. *Nature Methods* **6**, 263–265 (2009).

[26] Li, X. *et al.* Application of fuzzy c-means clustering in data analysis of metabolomics. *Analytical Chemistry* **81**, 4468–4475 (2009).

[27] Schaeffer, R. D. & Daggett, V. Protein folds and protein folding. *Protein Eng Des Sel* **24**, 11–9 (2011).

[28] Andreeva, A., Howorth, D., Chothia, C., Kulesha, E. & Murzin, A. G. Scop2 prototype: a new approach to protein structure mining. *Nucleic Acids Res* **42**, D310–4 (2014).

[29] Manning, G., Whyte, D. B., Martinez, R., Hunter, T. & Sudarsanam, S. The protein kinase complement of the human genome. *Science* **298**, 1912–34 (2002).

[30] Chen, M. J., Dixon, J. E. & Manning, G. Genomics and evolution of protein phosphatases. *Sci Signal* **10** (2017).

[31] Swint-Kruse, L. & Matthews, K. S. Allostery in the laci/galr family: variations on a theme. *Curr Opin Microbiol* **12**, 129–37 (2009).

[32] Reményi, A., Good, M. C. & Lim, W. A. Docking interactions in protein kinase and phosphatase networks. *Curr Opin Struct Biol* **16**, 676–85 (2006).

[33] Deribe, Y. L., Pawson, T. & Dikic, I. Post-translational modifications in signal integration. *Nat Struct Mol Biol* **17**, 666–72 (2010).

[34] Pawson, T. Organization of cell-regulatory systems through modular-protein-interaction domains. *Philos Trans A Math Phys Eng Sci* **361**, 1251–62 (2003).

[35] Lim, W. A. & Pawson, T. Phosphotyrosine signaling: evolving a new cellular communication system. *Cell* **142**, 661–7 (2010).

[36] Pincus, D., Letunic, I., Bork, P. & Lim, W. A. Evolution of the phospho-tyrosine signaling machinery in premetazoan lineages. *Proc Natl Acad Sci U S A* **105**, 9680–4 (2008).

[37] Finn, R. D. *et al.* The pfam protein families database: towards a more sustainable future. *Nucleic Acids Res* **44**, D279–85 (2016).

[38] Mitchell, A. *et al.* The interpro protein families database: the classification resource after 15 years. *Nucleic Acids Res* **43**, D213–21 (2015).

[39] Dobzhansky, T. Nothing in biology makes sense except in the light of evolution. *The American Biology Teacher* **75**, 87–91 (1973).

[40] Ohno, S. *Evolution by Gene Duplication* (Springer-Verlag, 1970).

[41] Raes, J. & Van de Peer, Y. Gene duplication, the evolution of novel gene functions, and detecting functional divergence of duplicates in silico. *Appl Bioinformatics* **2**, 91–101 (2003).

[42] Koonin, E. V. Orthologs, paralogs, and evolutionary genomics. *Annu Rev Genet* **39**, 309–38 (2005).

[43] Löytynoja, A. Alignment methods: strategies, challenges, benchmarking, and comparative overview. *Methods Mol Biol* **855**, 203–35 (2012).

[44] Chatzou, M. *et al.* Multiple sequence alignment modeling: methods and applications. *Brief Bioinform* **Epub ahead of print** (2015).

[45] Löytynoja, A. Phylogeny-aware alignment with prank. *Methods Mol Biol* **1079**, 155–70 (2014).

[46] Edgar, R. C. & Batzoglou, S. Multiple sequence alignment. *Curr Opin Struct Biol* **16**, 368–73 (2006).

[47] Notredame, C. Recent evolutions of multiple sequence alignment algorithms. *PLoS Comput Biol* **3**, e123 (2007).

[48] Do, C. B. & Katoh, K. Protein multiple sequence alignment. *Methods Mol Biol* **484**, 379–413 (2008).

[49] Pei, J. Multiple protein sequence alignment. *Curr Opin Struct Biol* **18**, 382–6 (2008).

[50] Kemena, C. & Notredame, C. Upcoming challenges for multiple sequence alignment methods in the high-throughput era. *Bioinformatics* **25**, 2455–65 (2009).

[51] Dessimoz, C. & Gil, M. Phylogenetic assessment of alignments reveals neglected tree signal in gaps. *Genome Biol* **11**, R37 (2010).

[52] Thompson, J. D., Linard, B., Lecompte, O. & Poch, O. A comprehensive benchmark study of multiple sequence alignment methods: current challenges and future perspectives. *PLoS One* **6**, e18093 (2011).

[53] Russell, D. J. (ed.) *Multiple Sequence Alignment Methods.* Methods in Molecular Biology (Humana Press, 2014).

[54] Henikoff, S. & Henikoff, J. G. Amino acid substitution matrices from protein blocks. *Proceedings of the National Academy of Sciences* **89**, 10915–10919 (1992).

[55] Dayhoff, M. O. & Schwartz, R. M. (eds.). *A model of evolutionary change in proteins*, chap. 22, 345–352. Atlas of Protein Sequence and Structure (National Biomedical Research Foundation, 1978).

[56] Gotoh, O. Heuristic alignment methods. *Methods Mol Biol* **1079**, 29–43 (2014).

[57] Needleman, S. B. & Wunsch, C. D. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J Mol Biol* **48**, 443–53 (1970).

[58] Gotoh, O. An improved algorithm for matching biological sequences. *J Mol Biol* **162**, 705–8 (1982).

[59] Hogeweg, P. & Hesper, B. The alignment of sets of sequences and the construction of phyletic trees: an integrated method. *J Mol Evol* **20**, 175–86 (1984).

[60] Feng, D. F. & Doolittle, R. F. Progressive sequence alignment as a prerequisite to correct phylogenetic trees. *J Mol Evol* **25**, 351–60 (1987).

[61] Katoh, K. & Standley, D. M. Mafft: iterative refinement and additional methods. *Methods Mol Biol* **1079**, 131–46 (2014).

[62] Liu, K., Linder, C. R. & Warnow, T. Multiple sequence alignment: a major challenge to large-scale phylogenetics. *PLoS Curr* **2**, RRN1198 (2010).

[63] Sievers, F., Dineen, D., Wilm, A. & Higgins, D. G. Making automated multiple alignments of very large numbers of protein sequences. *Bioinformatics* **29**, 989–95 (2013).

[64] Thompson, J. D., Koehl, P., Ripp, R. & Poch, O. Balibase 3.0: Latest developments of the multiple sequence alignment benchmark. *Proteins: Structure, Function, and Bioinformatics* **61**, 127–136 (2005).

[65] Stebbings, L. A. & Mizuguchi, K. Homstrad: recent developments of the homologous protein structure alignment database. *Nucleic Acids Res* **32**, D203–7 (2004).

[66] Bielawski, J. P. & Yang, Z. *Maximum Likelihood Methods for Detecting Adaptive Protein Evolution*, chap. 5, 103–124. Statistics for Biology and Health (Springer New York, 2005).

[67] Kosakovsky Pond, S. L. & Frost, S. D. W. Not so different after all: a comparison of methods for detecting amino acid sites under selection. *Mol Biol Evol* **22**, 1208–22 (2005).

[68] Massingham, T. Detecting the presence and location of selection in proteins. *Methods Mol Biol* **452**, 311–29 (2008).

[69] Williams, P. D., Pollock, D. D., Blackburne, B. P. & Goldstein, R. A. Assessing the accuracy of ancestral protein reconstruction methods. *PLoS Comput Biol* **2**, e69 (2006).

[70] Whelan, S. Inferring trees. *Methods Mol Biol* **452**, 287–309 (2008).

[71] Gonnet, G. H. Surprising results on phylogenetic tree building methods based on molecular sequences. *BMC Bioinformatics* **13**, 148 (2012).

[72] Hall, B. G. Comparison of the accuracies of several phylogenetic methods using protein and dna sequences. *Molecular Biology and Evolution* **22**, 792–802 (2005).

[73] Ogden, T. H. & Rosenberg, M. S. Multiple sequence alignment accuracy and phylogenetic inference. *Systematic Biology* **55**, 314–328 (2006).

[74] Wang, L.-S. . S. *et al.* The impact of multiple protein sequence alignment on phylogenetic estimation. In *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 4, 1108–1119. University of Texas at Austin, Austin (IEEE Computer Society, 2009).

[75] Yang, Z. *Computational molecular evolution* (Oxford University Press, Oxford, 2006).

[76] Warnow, T. *Large-Scale Multiple Sequence Alignment and Phylogeny Estimation*, chap. 6, 85–146 (Springer London, 2013).

[77] Löytynoja, A. & Goldman, N. Phylogeny-aware gap placement prevents errors in sequence alignment and evolutionary analysis. *Science* **320**, 1632–5 (2008).

[78] Morrison, D. A. Multiple sequence alignment for phylogenetic purposes. *AUSTRALIAN SYSTEMATIC BOTANY* **19**, 479–539 (2006).

[79] Morrison, D. A. Why would phylogeneticists ignore computerized sequence alignment? *Systematic Biology* **58**, 150–158 (2009).

[80] Iantorno, S., Gori, K., Goldman, N., Gil, M. & Dessimoz, C. *Who watches the watchmen? An appraisal of benchmarks for multiple sequence alignment.*, vol. 1079 of *Methods in Molecular Biology*, chap. 4, 59–73 (Springer International Publishing, United States, 2014).

[81] Saitou, N. & Nei, M. The neighbor-joining method: a new method for reconstructing phylogenetic trees. *Mol Biol Evol* **4**, 406–25 (1987).

[82] Gascuel, O. Bionj: an improved version of the nj algorithm based on a simple model of sequence data. *Mol Biol Evol* **14**, 685–95 (1997).

[83] Bruno, W. J., Socci, N. D. & Halpern, A. L. Weighted neighbor joining: a likelihood-based approach to distance-based phylogeny reconstruction. *Mol Biol Evol* **17**, 189–97 (2000).

[84] Liu, K., Raghavan, S., Nelesen, S., Linder, C. R. & Warnow, T. Rapid and accurate large-scale coestimation of sequence alignments and phylogenetic trees. *Science* **324**, 1561–4 (2009).

[85] Liu, K. *et al.* Sate-ii: very fast and accurate simultaneous estimation of multiple sequence alignments and phylogenetic trees. *Syst Biol* **61**, 90–106 (2012).

[86] Liu, K. & Warnow, T. *Large-scale multiple sequence alignment and tree estimation using SATé.*, vol. 1079 of *Methods in Molecular Biology*, chap. 15, 219–44 (Springer International Publishing, United States, 2014).

[87] Fletcher, W. & Yang, Z. The effect of insertions, deletions, and alignment errors on the branch-site test of positive selection. *Mol Biol Evol* **27**, 2257–67 (2010).

[88] Jordan, G. & Goldman, N. The effects of alignment error and alignment filtering on the sitewise detection of positive selection. *Mol Biol Evol* **29**, 1125–39 (2012).

[89] Capella-Gutiérrez, S. & Gabaldón, T. Measuring guide-tree dependency of inferred gaps in progressive aligners. *Bioinformatics* **29**, 1011–7 (2013).

[90] Redelings, B. D. & Suchard, M. A. Joint bayesian estimation of alignment and phylogeny. *Syst Biol* **54**, 401–18 (2005).

[91] Suchard, M. A. & Redelings, B. D. Bali-phy: simultaneous bayesian inference of alignment and phylogeny. *Bioinformatics* **22**, 2047–8 (2006).

[92] Novák, A., Miklós, I., Lyngsø, R. & Hein, J. Statalign: an extendable software package for joint bayesian estimation of alignments and evolutionary trees. *Bioinformatics* **24**, 2403–4 (2008).

[93] Arunapuram, P. *et al.* Statalign 2.0: combining statistical alignment with rna secondary structure prediction. *Bioinformatics* **29**, 654–5 (2013).

[94] Katoh, K. & Standley, D. M. Mafft multiple sequence alignment software version 7: Improvements in performance and usability. *Molecular Biology and Evolution* **30**, 772–780 (2013).

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

[96] Fitch, W. M. & Markowitz, E. 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**, 579–93 (1970).

[97] Penny, D., McComish, B. J., Charleston, M. A. & Hendy, M. D. Mathematical elegance with biochemical realism: the covarion model of molecular evolution. *J Mol Evol* **53**, 711–23 (2001).

[98] Zhou, Y., Brinkmann, H., Rodrigue, N., Lartillot, N. & Philippe, H. A dirichlet process covarion mixture model and its assessments using posterior predictive discrepancy tests. *Mol Biol Evol* **27**, 371–84 (2010).

[99] Göbel, U., Sander, C., Schneider, R. & Valencia, A. Correlated mutations and residue contacts in proteins. *Proteins* **18**, 309–17 (1994).

[100] Lockless, S. W. & Ranganathan, R. Evolutionarily conserved pathways of energetic connectivity in protein families. *Science* **286**, 295–9 (1999).

[101] Wollenberg, K. R. & Atchley, W. R. Separation of phylogenetic and functional associations in biological sequences by using the parametric bootstrap. *Proc Natl Acad Sci U S A* **97**, 3288–91 (2000).

[102] Kass, I. & Horovitz, A. Mapping pathways of allosteric communication in groel by analysis of correlated mutations. *Proteins* **48**, 611–7 (2002).

[103] Süel, G. M., Lockless, S. W., Wall, M. A. & Ranganathan, R. Evolutionarily conserved networks of residues mediate allosteric communication in proteins. *Nat Struct Biol* **10**, 59–69 (2003).

[104] Tillier, E. R. M. & Lui, T. W. H. Using multiple interdependency to separate functional from phylogenetic correlations in protein alignments. *Bioinformatics* **19**, 750–5 (2003).

[105] Fodor, A. A. & Aldrich, R. W. On evolutionary conservation of thermodynamic coupling in proteins. *J Biol Chem* **279**, 19046–50 (2004).

[106] Dekker, J. P., Fodor, A., Aldrich, R. W. & Yellen, G. A perturbation-based method for calculating explicit likelihood of evolutionary co-variance in multiple sequence alignments. *Bioinformatics* **20**, 1565–72 (2004).

[107] Fodor, A. A. & Aldrich, R. W. Influence of conservation on calculations of amino acid covariance in multiple sequence alignments. *Proteins* **56**, 211–21 (2004).

[108] Gloor, G. B., Martin, L. C., Wahl, L. M. & Dunn, S. D. Mutual information in protein multiple sequence alignments reveals two classes of coevolving positions. *Biochemistry* **44**, 7156–65 (2005).

[109] Socolich, M. *et al.* Evolutionary information for specifying a protein fold. *Nature* **437**, 512–8 (2005).

[110] Russ, W. P., Lowery, D. M., Mishra, P., Yaffe, M. B. & Ranganathan, R. Natural-like function in artificial ww domains. *Nature* **437**, 579–83 (2005).

[111] Martin, L. C., Gloor, G. B., Dunn, S. D. & Wahl, L. M. Using information theory to search for co-evolving residues in proteins. *Bioinformatics* **21**, 4116–24 (2005).

[112] Fares, M. A. & Travers, S. A. A. A novel method for detecting intramolecular coevolution: adding a further dimension to selective constraints analyses. *Genetics* **173**, 9–23 (2006).

[113] Yip, K. Y. *et al.* An integrated system for studying residue coevolution in proteins. *Bioinformatics* **24**, 290–2 (2008).

[114] Dunn, S. D., Wahl, L. M. & Gloor, G. B. Mutual information without the influence of phylogeny or entropy dramatically improves residue contact prediction. *Bioinformatics* **24**, 333–40 (2008).

[115] Weigt, M., White, R. A., Szurmant, H., Hoch, J. A. & Hwa, T. Identification of direct residue contacts in protein-protein interaction by message passing. *Proc Natl Acad Sci U S A* **106**, 67–72 (2009).

[116] Halabi, N., Rivoire, O., Leibler, S. & Ranganathan, R. Protein sectors: evolutionary units of three-dimensional structure. *Cell* **138**, 774–86 (2009).

[117] Gloor, G. B. *et al.* Functionally compensating coevolving positions are neither homoplasic nor conserved in clades. *Mol Biol Evol* **27**, 1181–91 (2010).

[118] Balakrishnan, S., Kamisetty, H., Carbonell, J. G., Lee, S.-I. I. & Langmead, C. J. Learning generative models for protein fold families. *Proteins* **79**, 1061–78 (2011).

[119] Morcos, F. *et al.* Direct-coupling analysis of residue coevolution captures native contacts across many protein families. *Proc Natl Acad Sci U S A* **108**, E1293–301 (2011).

[120] Jones, D. T., Buchan, D. W. A., Cozzetto, D. & Pontil, M. Psicov: precise structural contact prediction using sparse inverse covariance estimation on large multiple sequence alignments. *Bioinformatics* **28**, 184–90 (2012).

[121] Ashenberg, O. & Laub, M. T. Using analyses of amino acid coevolution to understand protein structure and function. *Methods Enzymol* **523**, 191–212 (2013).

[122] Casari, G., Sander, C. & Valencia, A. A method to predict functional residues in proteins. *Nat Struct Biol* **2**, 171–8 (1995).

[123] Lichtarge, O., Bourne, H. R. & Cohen, F. E. An evolutionary trace method defines binding surfaces common to protein families. *J Mol Biol* **257**, 342–58 (1996).

[124] Hannenhalli, S. S. & Russell, R. B. Analysis and prediction of functional sub-types from protein sequence alignments. *J Mol Biol* **303**, 61–76 (2000).

[125] Aloy, P., Querol, E., Aviles, F. X. & Sternberg, M. J. Automated structure-based prediction of functional sites in proteins: applications to assessing the validity of in-

heriting protein function from homology in genome annotation and to protein docking. *J Mol Biol* **311**, 395–408 (2001).

[126] Mirny, L. A. & Gelfand, M. S. Using orthologous and paralogous proteins to identify specificity-determining residues in bacterial transcription factors. *Journal of Molecular Biology* **321**, 7–20 (2002).

[127] del Sol Mesa, A., Pazos, F. & Valencia, A. Automatic methods for predicting functionally important residues. *J Mol Biol* **326**, 1289–302 (2003).

[128] Oliveira, L., Paiva, P. B., Paiva, A. C. M. & Vriend, G. Identification of functionally conserved residues with the use of entropy-variability plots. *Proteins* **52**, 544–52 (2003).

[129] Kalinina, O. V., Mironov, A. A., Gelfand, M. S. & Rakhmaninova, A. B. Automated selection of positions determining functional specificity of proteins by comparative analysis of orthologous groups in protein families. *Protein Sci* **13**, 443–56 (2004).

[130] Mihalek, I., Res, I. & Lichtarge, O. A family of evolution-entropy hybrid methods for ranking protein residues by importance. *J Mol Biol* **336**, 1265–82 (2004).

[131] Donald, J. E. & Shakhnovich, E. I. Determining functional specificity from protein sequences. *Bioinformatics* **21**, 2629–35 (2005).

[132] Pei, J., Cai, W., Kinch, L. N. & Grishin, N. V. Prediction of functional specificity determinants from protein sequences using log-likelihood ratios. *Bioinformatics* **22**, 164–71 (2006).

[133] Pirovano, W., Feenstra, K. A. & Heringa, J. Sequence comparison by sequence harmony identifies subtype-specific functional sites. *Nucleic Acids Res* **34**, 6540–8 (2006).

[134] Ye, K., Lameijer, E.-W. M. W., Beukers, M. W. & Ijzerman, A. P. A two-entropies analysis to identify functional positions in the transmembrane region of class a g protein-coupled receptors. *Proteins* **63**, 1018–30 (2006).

[135] Pazos, F., Rausell, A. & Valencia, A. Phylogeny-independent detection of functional residues. *Bioinformatics* **22**, 1440–8 (2006).

[136] Marttinen, P., Corander, J., Törönen, P. & Holm, L. Bayesian search of functionally divergent protein subgroups and their function specific residues. *Bioinformatics* **22**, 2466–74 (2006).

[137] Reva, B., Antipin, Y. & Sander, C. Determinants of protein function revealed by combinatorial entropy optimization. *Genome Biol* **8**, R232 (2007).

[138] Wallace, I. M. & Higgins, D. G. Supervised multivariate analysis of sequence groups to identify specificity determining residues. *BMC Bioinformatics* **8**, 135 (2007).

[139] Chakrabarti, S., Bryant, S. H. & Panchenko, A. R. Functional specificity lies within the properties and evolutionary changes of amino acids. *J Mol Biol* **373**, 801–10 (2007).

[140] Ye, K., Feenstra, K. A., Heringa, J., Ijzerman, A. P. & Marchiori, E. Multi-relief: a method to recognize specificity determining residues from multiple sequence alignments using a machine-learning approach for feature weighting. *Bioinformatics* **24**, 18–25 (2008).

[141] Ye, K., Vriend, G. & Ijzerman, A. P. Tracing evolutionary pressure. *Bioinformatics* **24**, 908–15 (2008).

[142] Capra, J. A. & Singh, M. Characterization and prediction of residues determining protein functional specificity. *Bioinformatics* **24**, 1473–80 (2008).

[143] Sankararaman, S. & Sjölander, K. Intrepid–information-theoretic tree traversal for protein functional site identification. *Bioinformatics* **24**, 2445–52 (2008).

[144] Chakrabarti, S. & Panchenko, A. R. Ensemble approach to predict specificity determinants: benchmarking and validation. *BMC Bioinformatics* **10**, 207 (2009).

[145] Brandt, B. W., Feenstra, K. A. & Heringa, J. Multi-harmony: detecting functional specificity from sequence alignment. *Nucleic Acids Res* **38**, W35–40 (2010).

[146] Marks, D. S. *et al.* Protein 3d structure computed from evolutionary sequence variation. *PLoS One* **6**, e28766 (2011).

[147] Marks, D. S., Hopf, T. A. & Sander, C. Protein structure prediction from sequence variation. *Nat Biotechnol* **30**, 1072–80 (2012).

[148] Ovchinnikov, S., Kamisetty, H. & Baker, D. Robust and accurate prediction of residue-residue interactions across protein interfaces using evolutionary information. *Elife* **3**, e02030 (2014).

[149] Reynolds, K. A., McLaughlin, R. N. & Ranganathan, R. Hot spots for allosteric regulation on protein surfaces. *Cell* **147**, 1564–75 (2011).

[150] McLaughlin, R. N., Poelwijk, F. J., Raman, A., Gosal, W. S. & Ranganathan, R. The spatial architecture of protein function and adaptation. *Nature* **491**, 138–42 (2012).

[151] Valdar, W. S. J. Scoring residue conservation. *Proteins* **48**, 227–41 (2002).

[152] Capra, J. A. & Singh, M. Predicting functionally important residues from sequence conservation. *Bioinformatics* **23**, 1875–82 (2007).

[153] Fischer, J. D., Mayer, C. E. & Söding, J. Prediction of protein functional residues from sequence by probability density estimation. *Bioinformatics* **24**, 613–20 (2008).

[154] Löytynoja, A., Vilella, A. J. & Goldman, N. Accurate extension of multiple sequence alignments using a phylogeny-aware graph algorithm. *Bioinformatics* **28**, 1684–91 (2012).

[155] Waterman, M. S. Sequence alignments in the neighborhood of the optimum with general application to dynamic programming. *Proceedings of the National Academy of Sciences* **80**, 3123–3124 (1983).

[156] Vingron, M. Near-optimal sequence alignment. *Curr Opin Struct Biol* **6**, 346–52 (1996).

[157] Landan, G. & Graur, D. Heads or tails: a simple reliability check for multiple sequence alignments. *Mol Biol Evol* **24**, 1380–3 (2007).

[158] Wong, K. M., Suchard, M. A. & Huelsenbeck, J. P. Alignment uncertainty and genomic analysis. *Science* **319**, 473–6 (2008).

[159] Morrison, D. A. & Ellis, J. T. Effects of nucleotide sequence alignment on phylogeny estimation: a case study of 18s rdnas of apicomplexa. *Mol Biol Evol* **14**, 428–41 (1997).

[160] Mugridge, N. B. *et al.* Effects of sequence alignment and structural domains of ribosomal dna on phylogeny reconstruction for the protozoan family sarcocystidae. *Molecular Biology and Evolution* **17**, 1842–1853 (2000).

[161] Cantarel, B. L., Morrison, H. G. & Pearson, W. Exploring the relationship between sequence similarity and accurate phylogenetic trees. *Molecular Biology and Evolution* **23**, 2090–2100 (2006).

[162] Castresana, J. Selection of conserved blocks from multiple alignments for their use in phylogenetic analysis. *Mol Biol Evol* **17**, 540–52 (2000).

[163] Löytynoja, A. & Milinkovitch, M. C. Soap, cleaning multiple alignments from unstable blocks. *Bioinformatics* **17**, 573–4 (2001).

[164] Penn, O., Privman, E., Landan, G., Graur, D. & Pupko, T. An alignment confidence score capturing robustness to guide tree uncertainty. *Mol Biol Evol* **27**, 1759–67 (2010).

[165] Kück, P. *et al.* Parametric and non-parametric masking of randomness in sequence alignments can be improved and leads to better resolved trees. *Front Zool* **7**, 10 (2010).

[166] Wu, M., Chatterji, S. & Eisen, J. A. Accounting for alignment uncertainty in phylogenomics. *PLoS One* **7**, e30288 (2012).

[167] Cho, Y., Sloutsky, R., Naegle, K. M. & Cavalli, V. Injury-induced hdac5 nuclear export is essential for axon regeneration. *Cell* **155**, 894–908 (2013).

[168] Liu, K., Tedeschi, A., Park, K. K. & He, Z. Neuronal intrinsic mechanisms of axon regeneration. *Annu Rev Neurosci* **34**, 131–52 (2011).

[169] Smith, D. S. & Skene, J. H. A transcription-dependent switch controls competence of adult neurons for distinct modes of axon growth. *J Neurosci* **17**, 646–58 (1997).

[170] Tedeschi, A. Tuning the orchestra: transcriptional pathways controlling axon regeneration. *Front Mol Neurosci* **4**, 60 (2011).

[171] McQuarrie, I. G. & Grafstein, B. Axon outgrowth enhanced by a previous nerve injury. *Arch Neurol* **29**, 53–5 (1973).

[172] Richardson, P. M. & Issa, V. M. Peripheral injury enhances central regeneration of primary sensory neurones. *Nature* **309**, 791–3 (1984).

[173] Vega, R. B. *et al.* Protein kinases c and d mediate agonist-dependent cardiac hypertrophy through nuclear export of histone deacetylase 5. *Mol Cell Biol* **24**, 8374–85 (2004).

[174] Chawla, S., Vanhoutte, P., Arnold, F. J. L., Huang, C. L.-H. & Bading, H. Neuronal activity-dependent nucleocytoplasmic shuttling of hdac4 and hdac5. *J Neurochem* **85**, 151–9 (2003).

[175] Cho, Y. & Cavalli, V. Hdac5 is a novel injury-regulated tubulin deacetylase controlling axon regeneration. *EMBO J* **31**, 3063–78 (2012).

[176] Ha, C. H. *et al.* Pka phosphorylates histone deacetylase 5 and prevents its nuclear export, leading to the inhibition of gene transcription and cardiomyocyte hypertrophy. *Proc Natl Acad Sci U S A* **107**, 15467–72 (2010).

[177] Leah, J. D., Herdegen, T., Murashov, A., Dragunow, M. & Bravo, R. Expression of immediate early gene proteins following axotomy and inhibition of axonal transport in the rat central nervous system. *Neuroscience* **57**, 53–66 (1993).

[178] Broude, E., McAtee, M., Kelley, M. S. & Bregman, B. S. c-jun expression in adult rat dorsal root ganglion neurons: differential response after central or peripheral axotomy. *Exp Neurol* **148**, 367–77 (1997).

[179] Cayrou, C., Denver, R. J. & Puymirat, J. Suppression of the basic transcription element-binding protein in brain neuronal cultures inhibits thyroid hormone-induced neurite branching. *Endocrinology* **143**, 2242–9 (2002).

[180] Moore, D. L. *et al.* Klf family members regulate intrinsic axon regeneration ability. *Science* **326**, 298–301 (2009).

[181] Buschmann, T. *et al.* Expression of jun, fos, and atf-2 proteins in axotomized explanted and cultured adult rat dorsal root ganglia. *Neuroscience* **84**, 163–76 (1998).

[182] Xiong, X. *et al.* Protein turnover of the wallenda/dlk kinase regulates a retrograde response to axonal injury. *J Cell Biol* **191**, 211–23 (2010).

[183] Befort, K., Karchewski, L., Lanoue, C. & Woolf, C. J. Selective up-regulation of the growth arrest dna damage-inducible gene gadd45 alpha in sensory and motor neurons after peripheral nerve injury. *Eur J Neurosci* **18**, 911–22 (2003).

[184] Michaelevski, I. *et al.* Signaling to transcription networks in the neuronal retrograde injury response. *Sci Signal* **3**, ra53 (2010).

[185] Blesch, A. *et al.* Conditioning lesions before or after spinal cord injury recruit broad genetic mechanisms that sustain axonal regeneration: superiority to camp-mediated effects. *Exp Neurol* **235**, 162–73 (2012).

[186] Gaub, P. *et al.* Hdac inhibition promotes neuronal outgrowth and counteracts growth cone collapse through cbp/p300 and p/caf-dependent p53 acetylation. *Cell Death Differ* **17**, 1392–408 (2010).

[187] Gaub, P. *et al.* The histone acetyltransferase p300 promotes intrinsic axonal regeneration. *Brain* **134**, 2134–48 (2011).

[188] Riccio, A. Dynamic epigenetic regulation in neurons: enzymes, stimuli and signaling pathways. *Nat Neurosci* **13**, 1330–7 (2010).

[189] McKinsey, T. A., Zhang, C. L. & Olson, E. N. Activation of the myocyte enhancer factor-2 transcription factor by calcium/calmodulin-dependent protein kinase-stimulated binding of 14-3-3 to histone deacetylase 5. *Proc Natl Acad Sci U S A* **97**, 14400–5 (2000).

[190] Iskandar, B. J. *et al.* Folate regulation of axonal regeneration in the rodent central nervous system through dna methylation. *J Clin Invest* **120**, 1603–16 (2010).

[191] Abe, N., Borson, S. H., Gambello, M. J., Wang, F. & Cavalli, V. Mammalian target of rapamycin (mtor) activation increases axonal growth capacity of injured peripheral nerves. *J Biol Chem* **285**, 28034–43 (2010).

[192] Bolstad, B. M., Irizarry, R. A., Astrand, M. & Speed, T. P. A comparison of normalization methods for high density oligonucleotide array data based on variance and bias. *Bioinformatics* **19**, 185–93 (2003).

[193] Baldi, P. & Long, A. D. A bayesian framework for the analysis of microarray expression data: regularized t -test and statistical inferences of gene changes. *Bioinformatics* **17**, 509–19 (2001).

[194] Naegle, K. M. *et al.* Ptmscout, a web resource for analysis of high throughput post-translational proteomics studies. *Mol Cell Proteomics* **9**, 2558–70 (2010).

[195] Sloutsky, R., Jimenez, N., Swamidass, S. J. & Naegle, K. M. Accounting for noise when clustering biological data. *Brief Bioinform* **14**, 423–36 (2013).

[196] Jain, A. K., Murty, N. M. & Flynn, P. J. Data Clustering : A Review. *ACM Computing Surveys* **31**, 264–323 (1999).

[197] Wolf-Yadlin, A. *et al.* Effects of HER2 overexpression on cell signaling networks governing proliferation and migration. *Molecular systems biology* **2**, 54 (2006).

[198] Lee, M. L., Kuo, F. C., Whitmore, G. a. & Sklar, J. Importance of replication in microarray gene expression studies: statistical methods and evidence from repetitive cDNA hybridizations. *Proceedings of the National Academy of Sciences of the United States of America* **97**, 9834–9 (2000).

[199] Dougherty, E. R. *et al.* Inference from Clustering with Application to Gene-Expression Microarrays. *Journal of Computational Biology* **9**, 105–126 (2002).

[200] Medvedovic, M., Yeung, K. & Bumgarner, R. Bayesian mixture model based clustering of replicated microarray data. *Bioinformatics* **20**, 1222–1232 (2004).

[201] Ng, S. K., McLachlan, G. J., Wang, K., Ben-Tovim Jones, L. & Ng, S.-W. A mixture model with random-effects components for clustering correlated gene-expression profiles. *Bioinformatics (Oxford, England)* **22**, 1745–52 (2006).

[202] Cooke, E. J., Savage, R. S., Kirk, P. D. W., Darkins, R. & Wild, D. L. Bayesian hierarchical clustering for microarray time series data with replicates and outlier measurements. *BMC bioinformatics* **12**, 399 (2011).

[203] Yeung, K. Y., Medvedovic, M. & Bumgarner, R. E. Clustering gene-expression data with repeated measurements. *Genome Biology* **4** (2003).

[204] Bittner, M. *et al.* Molecular classification of cutaneous malignant melanoma by gene expression profiling. *Nature* **406**, 536–540 (2000).

[205] Strehl, A. & Gosh, J. Cluster Ensembles – A Knowledge Reuse Framework for Combining Multiple Partitions. *Journal of Machine Learning Research* **3**, 583–617 (2002).

[206] Topchy, A., Jain, A. K. & Punch, W. Clustering ensembles: models of consensus and weak partitions. *IEEE transactions on pattern analysis and machine intelligence* **27**, 1866–81 (2005).

[207] Grotkjaer, T., Winther, O., Regenberg, B., Nielsen, J. & Hansen, L. K. Robust multi-scale clustering of large DNA microarray datasets with the consensus algorithm. *Bioinformatics (Oxford, England)* **22**, 58–67 (2006).

[208] Monti, S., Tamayo, P., Mesirov, J. & Golub, T. Consensus Clustering : A Resampling-Based Method for Class Discovery and Visualization of Gene Expression Microarray Data. *Machine Learning* **52**, 91–118 (2003).

[209] Fred, A. Finding consistent clusters in data partitions. In Kittler, J. & Roli, F. (eds.) *Multiple Classifier Systems*, 309–318 (Springer, 2001), lncs 2096 edn.

[210] Kuncheva, L. I. & Vetrov, D. P. Evaluation of stability of k-means cluster ensembles with respect to random initialization. *IEEE transactions on pattern analysis and machine intelligence* **28**, 1798–808 (2006).

[211] Naegle, K. M., Welsch, R. E., Yaffe, M. B., White, F. M. & Lauffenburger, D. A. MCAM: Multiple Clustering Analysis Methodology for Deriving Hypotheses and Insights from High-Throughput Proteomic Datasets. *PLoS computational biology* **7** (2011).

[212] Bellec, P., Rosa-Neto, P., Lyttelton, O. C., Benali, H. & Evans, A. C. Multi-level bootstrap analysis of stable clusters in resting-state fMRI. *NeuroImage* **51**, 1126–39 (2010).

[213] Iam-on, N. & Garrett, S. LinkCluE : A MATLAB Package for Link-Based Cluster Ensembles. *Journal Of Statistical Software* **36** (2010).

[214] Avogadri, R. & Valentini, G. Fuzzy ensemble clustering based on random projections for DNA microarray data analysis. *Artificial intelligence in medicine* **45**, 173–83 (2009).

[215] Mimaroglu, S. & Yagci, M. CLICOM: Cliques for combining multiple clusterings. *Expert Systems With Applications* **39**, 1889–1901 (2012).

[216] Baldi, P. & Long, A. D. A Bayesian framework for the analysis of microarray expression data: regularized t-test and statistical inferences of gene changes. *Bioinformatics* **17**, 509–519 (2001).

[217] Hampel, F. R., Ronchetti, E. M., Rousseeuw, P. J. & Stahel, W. A. *Robust Statistics: The Approach Based on Influence Functions* (John Wiley & Sons, New York, 1986).

[218] Draghici, S. *et al.* Noise sampling method: an ANOVA approach allowing robust selection of differentially regulated genes measured by DNA microarrays. *Bioinformatics* **19**, 1348–1359 (2003).

[219] Schmelzle, K., Kane, S., Gridley, S., Lienhard, G. E. & White, F. M. Temporal Dynamics of Tyrosine Phosphorylation in Insulin Signaling. *Diabetes* **55**, 2171–2179 (2006).

[220] Amit, I. *et al.* A module of negative feedback regulators defines growth factor signaling. *Nat Genet* **39**, 503–512 (2007).

[221] Carson, J. P. *et al.* Pharmacogenomic identification of targets for adjuvant therapy with the topoisomerase poison camptothecin. *Cancer Res* **64**, 2096–2104 (2004).

[222] Viegas, M. H., Gehring, N. H., Breit, S., Hentze, M. W. & Kulozik, A. E. The abundance of rnps1, a protein component of the exon junction complex, can determine the variability in efficiency of the nonsense mediated decay pathway. *Nucleic Acids Res* **35**, 4542–4551 (2007).

[223] Giles, P. J. & Kipling, D. Normality of oligonucleotide microarray data and implications for parametric statistical analyses. *Bioinformatics* **19**, 2254–62 (2003).

[224] Tu, Y., Stolovitzky, G. & Klein, U. Quantitative noise analysis for gene expression microarray experiments. *Proc Natl Acad Sci U S A* **99**, 14031–14036 (2002).

[225] Hardin, J. & Wilson, J. A note on oligonucleotide expression values not being normally distributed. *Biostatistics* **10**, 446–450 (2009).

[226] Sloutsky, R. & Naegle, K. M. High-resolution identification of specificity determining positions in the laci protein family using ensembles of sub-sampled alignments. *PLoS One* **11**, e0162579 (2016).

[227] Curwen, V. *et al.* The ensembl automatic gene annotation system. *Genome Res* **14**, 942–50 (2004).

[228] Loewenstein, Y. *et al.* Protein function annotation by homology-based inference. *Genome Biol* **10**, 207 (2009).

[229] Richardson, E. J. & Watson, M. The automatic annotation of bacterial genomes. *Brief Bioinform* **14**, 1–12 (2013).

[230] Kimura, N. *The Neutral Theory of Molecular Evolution* (Cambridge University Press, 1983).

[231] Ashkenazy, H. *et al.* Consurf 2016: an improved methodology to estimate and visualize evolutionary conservation in macromolecules. *Nucleic Acids Res* **44**, W344–50 (2016).

[232] Meinhardt, S. *et al.* Novel insights from hybrid laci/galr proteins: family-wide functional attributes and biologically significant variation in transcription repression. *Nucleic Acids Res* **40**, 11139–54 (2012).

[233] Sousa, F. L. *et al.* Allorep: a repository of sequence, structural and mutagenesis data for the laci/galr transcription regulators. *J Mol Biol* **428**, 671–8 (2015).

[234] Prince, V. E. & Pickett, F. B. Splitting pairs: the diverging fates of duplicated genes. *Nat Rev Genet* **3**, 827–37 (2002).

[235] Schneider, T. D. & Stephens, R. M. Sequence logos: a new way to display consensus sequences. *Nucleic Acids Res* **18**, 6097–100 (1990).

[236] Crooks, G. E., Hon, G., Chandonia, J.-M. & Brenner, S. E. Weblogo: a sequence logo generator. *Genome Res* **14**, 1188–90 (2004).

[237] Kersey, P. J. *et al.* Ensembl genomes 2016: more genomes, more complexity. *Nucleic Acids Research* **44**, D574–D580 (2016). `http://nar.oxfordjournals.org/content/44/D1/D574.full.pdf+html`.

[238] Katoh, K., Kuma, K.-i. . I., Toh, H. & Miyata, T. Mafft version 5: improvement in accuracy of multiple sequence alignment. *Nucleic Acids Research* **33**, 511–518 (2005).

[239] Konagurthu, A. S., Whisstock, J. C., Stuckey, P. J. & Lesk, A. M. Mustang: a multiple structural alignment algorithm. *Proteins* **64**, 559–74 (2006).

[240] Espinosa-Cantú, A., Ascencio, D., Barona-Gómez, F. & DeLuna, A. Gene duplication and the evolution of moonlighting proteins. *Front Genet* **6**, 227 (2015).

[241] Yang, Z. & Nielsen, R. Codon-substitution models for detecting molecular adaptation at individual sites along specific lineages. *Mol Biol Evol* **19**, 908–17 (2002).

[242] Massingham, T. & Goldman, N. Detecting amino acid sites under positive selection and purifying selection. *Genetics* **169**, 1753–62 (2005).

[243] Harms, M. J. & Thornton, J. W. Analyzing protein structure and function using ancestral gene reconstruction. *Curr Opin Struct Biol* **20**, 360–6 (2010).

[244] Pupko, T., Bell, R. E., Mayrose, I., Glaser, F. & Ben-Tal, N. Rate4site: an algorithmic tool for the identification of functional regions in proteins by surface mapping of evo-

lutionary determinants within their homologues. *Bioinformatics* **18 Suppl 1**, S71–7 (2002).

[245] Dutheil, J. Y. Detecting coevolving positions in a molecule: why and how to account for phylogeny. *Brief Bioinform* **13**, 228–43 (2012).

[246] Thomson, J. M. *et al.* Resurrecting ancestral alcohol dehydrogenases from yeast. *Nat Genet* **37**, 630–5 (2005).

[247] Bridgham, J. T. *et al.* Protein evolution by molecular tinkering: diversification of the nuclear receptor superfamily from a ligand-dependent ancestor. *PLoS Biol* **8** (2010).

[248] Eick, G. N., Colucci, J. K., Harms, M. J., Ortlund, E. A. & Thornton, J. W. Evolution of minimal specificity and promiscuity in steroid hormone receptors. *PLoS Genet* **8**, e1003072 (2012).

[249] Baker, C. R., Hanson-Smith, V. & Johnson, A. D. Following gene duplication, paralog interference constrains transcriptional circuit evolution. *Science* **342**, 104–8 (2013).

[250] Rahman, T. *et al.* Two-pore channels provide insight into the evolution of voltage-gated ca2+ and na+ channels. *Sci Signal* **7**, ra109 (2014).

[251] Creixell, P. *et al.* Unmasking determinants of specificity in the human kinome. *Cell* **163**, 187–201 (2015).

[252] Liu, B. A. *et al.* The human and mouse complement of sh2 domain proteins-establishing the boundaries of phosphotyrosine signaling. *Mol Cell* **22**, 851–68 (2006).

[253] Blackburne, B. P. & Whelan, S. Class of multiple sequence alignment algorithm affects genomic analysis. *Mol Biol Evol* **30**, 642–53 (2013).

[254] Shen, X.-X., Hittinger, C. T. & Rokas, A. Contentious relationships in phylogenomic studies can be driven by a handful of genes. *Nature Ecology & Evolution* **1**, 0126 (2017).

[255] Salichos, L. & Rokas, A. Inferring ancient divergences requires genes with strong phylogenetic signals. *Nature* **497**, 327–31 (2013).

[256] Sievers, F. *et al.* Fast, scalable generation of high-quality protein multiple sequence alignments using clustal omega. *Mol Syst Biol* **7**, 539 (2011).

[257] Edgar, R. C. Muscle: multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Res* **32**, 1792–7 (2004).

[258] Price, M. N., Dehal, P. S. & Arkin, A. P. Fasttree 2–approximately maximum-likelihood trees for large alignments. *PLoS One* **5**, e9490 (2010).

[259] Stamatakis, A. Raxml version 8: a tool for phylogenetic analysis and post-analysis of large phylogenies. *Bioinformatics* **30**, 1312–3 (2014).

[260] Robinson, D. F. & Foulds, L. R. Comparison of phylogenetic trees. *Mathematical biosciences* **53**, 131–147 (1981).

[261] Liu, K., Linder, C. R. & Warnow, T. Raxml and fasttree: comparing two methods for large-scale maximum likelihood phylogeny estimation. *PLoS One* **6**, e27731 (2011).

[262] Aken, B. L. *et al.* Ensembl 2017. *Nucleic Acids Res* **45**, D635–D642 (2017).

[263] Benson, D. A. *et al.* Genbank. *Nucleic Acids Res* **45**, D37–D42 (2017).

[264] Boc, A., Diallo, A. B. & Makarenkov, V. T-rex: a web server for inferring, validating and visualizing phylogenetic trees and networks. *Nucleic Acids Res* **40**, W573–9 (2012).

[265] Kuhner, M. K. & Felsenstein, J. A simulation comparison of phylogeny algorithms under equal and unequal evolutionary rates. *Mol Biol Evol* **11**, 459–68 (1994).

[266] Herrero, J. *et al.* Ensembl comparative genomics resources. *Database (Oxford)* **2016** (2016).

[267] Hedges, S. B., Marin, J., Suleski, M., Paymer, M. & Kumar, S. Tree of life reveals clock-like speciation and diversification. *Mol Biol Evol* **32**, 835–45 (2015).

[268] Kumar, S., Stecher, G., Suleski, M. & Hedges, S. B. Timetree: A resource for timelines, timetrees, and divergence times. *Mol Biol Evol* **34**, 1812–1819 (2017).

[269] Strope, C. L., Scott, S. D. & Moriyama, E. N. indel-seq-gen: a new protein family simulator incorporating domains, motifs, and indels. *Mol Biol Evol* **24**, 640–9 (2007).

[270] Strope, C. L., Abel, K., Scott, S. D. & Moriyama, E. N. Biological sequence simulation for testing complex evolutionary hypotheses: indel-seq-gen version 2.0. *Mol Biol Evol* **26**, 2581–93 (2009).

[271] Rambaut, A. & Grassly, N. C. Seq-gen: an application for the monte carlo simulation of dna sequence evolution along phylogenetic trees. *Comput Appl Biosci* **13**, 235–8 (1997).

[272] Liu, B. A. *et al.* Sh2 domains recognize contextual peptide sequence information to determine selectivity. *Mol Cell Proteomics* **9**, 2391–404 (2010).

[273] Hause, R. J. *et al.* Comprehensive binary interaction mapping of sh2 domains via fluorescence polarization reveals novel functional diversification of erbb receptors. *PLoS One* **7**, e44471 (2012).