Chapter 6

Computational Prediction of Short Linear Motifs from Protein Sequences

Richard J. Edwards and Nicolas Palopoli

Abstract

Short Linear Motifs (SLiMs) are functional protein microdomains that typically mediate interactions between a short linear region in one protein and a globular domain in another. SLiMs usually occur in structurally disordered regions and mediate low affinity interactions. Most SLiMs are 3–15 amino acids in length and have 2–5 defined positions, making them highly likely to occur by chance and extremely difficult to identify. Nevertheless, our knowledge of SLiMs and capacity to predict them from protein sequence data using computational methods has advanced dramatically over the past decade. By considering the biological, structural, and evolutionary context of SLiM occurrences, it is possible to differentiate functional instances from chance matches in many cases and to identify new regions of proteins that have the features consistent with a SLiM-mediated interaction. Their simplicity also makes SLiMs evolutionarily labile and prone to independent origins on different sequence backgrounds through convergent evolution, which can be exploited for predicting novel SLiMs in proteins that share a function or interaction partner.

In this review, we explore our current knowledge of SLiMs and how it can be applied to the task of predicting them computationally from protein sequences. Rather than focusing on specific SLiM prediction tools, we provide an overview of the methods available and concentrate on principles that should continue to be paramount even in the light of future developments. We consider the relative merits of using regular expressions or profiles for SLiM discovery and discuss the main considerations for both predicting new instances of known SLiMs, and de novo prediction of novel SLiMs. In particular, we highlight the importance of correctly modelling evolutionary relationships and the probability of false positive predictions.

Key words Short linear motifs, SLiM, Motif discovery, Protein-protein interactions, Posttranslational modifications, Intrinsically disordered proteins, Regular expressions, Sequence profiles, Sequence motifs

Abbreviations

DMI	Domain-motif interaction
ELM	Eukaryotic linear motif
FPR	False positive rate
GO	Gene ontology
HMM	Hidden Markov model
IDP	Intrinsically disordered protein

Peng Zhou and Jian Huang (eds.), *Computational Peptidology*, Methods in Molecular Biology, vol. 1268,
DOI 10.1007/978-1-4939-2285-7_6, © Springer Science+Business Media New York 2015

IDR	Intrinsically disordered region
LDMS	(*l, d*) motif search
MnM	Minimotif miner
MoRF	Molecular recognition feature
MST	Minimum spanning tree
PPI	Protein-protein interaction
PSSM	Position-specific scoring matrix
PTM	Posttranslational modification
Regex	Regular expression
SLiM	Short linear motif
TPR	True positive rate

1 Introduction

Short Linear Motifs (SLiMs) are a set of protein sequence features with specific attributes that the name suggests [1]:

1. *Short*. SLiMs are typically 3–15 amino acids in length, often with fewer than six (and as few as two) residues that are key to the function.
2. *Linear*. SLiMs are found in linear stretches of protein, typically in intrinsically disordered regions (IDR), and their (unbound) three-dimensional structure is not considered crucial for their activity.
3. *Motif*. SLiMs contain specific residues that are important for function and, as such, are amenable to sequence analysis tools and representations.

The functional relevance of short linear peptides has been recognized for decades (e.g., [2, 3]) but it was only in the early twenty-first century that SLiMs were recognized as a discrete class of element worthy of study in its own right [4, 5]. SLiMs are now recognized to be one of the key components in the cell's repertoire of protein-protein interactions (PPI), mediating a specific type [6] that we refer to here as a domain-motif interaction (DMI). Although it is hard to make a good estimate, it has been suggested that something in the order of 15–40 % of the PPI in a cell may be DMI [7]—a number which is likely to be enriched in signalling networks [8]. In the 10 years that the Eukaryotic Linear Motif (ELM) database has been collecting and curating SLiMs, the number of distinct classes has increased from approx. 80 in 2003 [5] to approx. 200 in 2013 [9] and is set to continue to rise. The latest release of Minimotif Miner (MnM) includes 880 consensus SLiMs [10], although this number is somewhat inflated by the way that length variability and redundancy is handled in the database. This suggests that even though progress in the field has led to the accumulation of much data on SLiMs, there is still much room for discovery of new instances of known and novel motifs.

SLiMs are involved in an incredibly diverse range of biological processes, including cell signalling, posttranslational modification, subcellular localisation, gene expression, membrane binding, protein folding, and cell adhesion [1, 8, 9, 11–14]. SLiMs usually bind with low affinity [8], making them ideal components to establish quick or transient responses. However, many motifs (of the same or different type) can co-occur, acting synergistically to give higher binding affinities [6, 8]. SLiMs also play an important role in disease; not only are they involved in core biological processes that can affect health if they go wrong but the evolutionary plasticity of SLiMs makes them ideal targets for exploitation by viruses via convergent evolution [15, 16].

Methods for SLiM prediction are under constant refinement and development and so this review is neither intended as an explicit "how to" guide to SLiM discovery nor an exhaustive list of methods and tools. Instead, we give an overview of the considerations that need to be made during such analyses, with examples from the literature, and some thoughts on future developments. This review highlights a selection of tools that illustrate key aspects of computational SLiM discovery. A particular focus is given to the tools of the SLiMSuite package, which is specifically geared to the analysis of SLiMs, including some tools that have not previously been published (Table 1). Additional SLiM prediction tools can be found in reviews by Diella et al. [8] and Davey et al. [11].

1.1 SLiM Terminology

The terminology related to SLiM analyses can be confusing because it uses a number of different terms from both biology and computing, some of which have developed their own SLiM-specific meanings. The main terms used in this chapter are therefore listed in the glossary (Table 2; *see* also [8]). We have made every effort to be consistent within the chapter but readers should be aware that some of the terms used can have alternative meanings in related disciplines. The term "motif" is particularly widespread and has a number of discipline-specific meanings. Within this review, "motif" refers to a short sequence motif unless otherwise specified.

1.2 SLiM Notation

A standard notation has been suggested to denote SLiMs in the written literature; *see* [4]. This is not universally applied and variation in notation can be found among publications, even when they describe the same motif [11]. Instead, there are two main classes of motif representation that are commonly used for computational analysis of SLiMs: regular expressions and sequence profiles, referred to in later sections simply as "regex" and "profile", respectively. The former are simple human- and machine-readable qualitative representations of which amino acids are tolerated at which positions in the SLiM. The latter, which for the purposes of this review includes position (specific) scoring/weight matrices

Table 1
The main tools of the SLiMSuite bioinformatics package

Tool	Ref.	Description	Web[a]
Compari Motif	[69]	A unique motif-motif comparison tool for identifying similar SLiMs. Used for clustering results of predictions and identifying known motifs.	Y
GABLAM	[77]	BLAST-based protein similarity scoring and clustering. Used for (Q)SLiMFinder and SLiMProb adjustments for evolutionary relationships.	N
GOPHER	[42]	Automated orthologue prediction and alignment algorithm. Used for conservation-based masking ((Q)SLiMFinder/SLiMProb) and prediction (SLiMPrints).	Y
PRESTO	[b]	Forerunner of SLiMSearch (now SLiMProb). A tool for searching predefined SLiMs against a protein dataset. Does not include overrepresentation/underrepresentation statistics but allows mismatches and more flexible SLiM definitions.	N
QSLiM Finder	[41][b]	Query-based variant of SLiMFinder with increased sensitivity and specificity, ideal for SLiM discovery from host-pathogen interactions or where at least one interaction is established experimentally.	N
SLiMBench	[b]	A new tool for creating and assessing de novo SLiM prediction benchmarking datasets.	N
SLiMdb	[63]	Interactive web pages to explore results of interactome-wide de novo SLiM prediction in humans, with links to other SLiMSuite tools and online public resources.	Y[c]
SLiMDisc	[42, 77]	One of the first de novo SLiM prediction tools that corrected for evolutionary relationships. Based on heuristic ranking of overrepresented motifs in unrelated proteins.	Y
SLiMFinder	[41, 100]	The first de novo SLiM prediction based on a statistical model of overrepresented motifs in unrelated proteins. Repeatedly achieves the greatest specificity in benchmarking.	Y
SLiMMaker	[b]	A simple tool for converting aligned peptides or SLiM occurrences into a regular expression motif.	Y
SLiMPred	[129]	Machine Learning de novo SLiM/MoRF prediction in single proteins based on known motif attributes.	Y[c]
SLiMPrints	[97]	Novel de novo SLiM/MoRF prediction in single proteins from statistical clustering of conserved disordered residues.	Y[c]
SLiMProb	[25]	Unique tool providing biological context (disorder and conservation) for searches of predefined SLiMs along with underrepresentation and overrepresentation statistics, correcting for evolutionary relationships. Formerly called SLiMSearch 1.x but renamed to avoid confusion with SLiMSearch2.	Y
SLiM Search2	[68]	Advanced biological context (disorder, conservation, and protein features), and ranking for proteome-wide searches of predefined motifs. Provides simple enrichment statistics for PPI partners and GO terms.	Y[c]

[a] Webserver available at http://bioware.ucd.ie/
[b] Not published at time of press. See citation details at: http://bioware.soton.ac.uk/
[c] Webserver only. Not part of SLiMSuite download

Table 2
Glossary of key terms

Term	Related terms	Description
Convergent evolution	Molecular mimicry	Independent evolutionary origins of the same function or motif on different genetic backgrounds.
Degenerate	Ambiguous	A SLiM position that can have 2+ different amino acids.
Divergent evolution	Conservation	The accumulation of differences over time following shared ancestry. Where such differences are selected against (purifying selection) sequence conservation will be seen.
Domain-Motif Interaction	DMI	PPI mediated by a SLiM in one protein and a SLiM-binding domain in the other.
Intrinsically Disordered Protein/Region	IDP/IDR	A protein/region that lacks a stable three-dimensional structure in the unbound state.
Instance	Occurrence	A single observation of a SLiM in a single protein.
(*l*, *d*) Motif Search	LDMS, (*l*, *d*) challenge problem, planted motif search	Motif search algorithms that search for recurring motifs of total length *l* with up to *d* mismatches in each occurrence.
MoRF	MoRE	Molecular Recognition Feature/Element. Short to medium-length, intrinsically disordered protein regions that mediate PPI via disorder-to-order transitions.
Pattern	Motif definition	The regular expression that defines a motif.
Posttranslational Modification	PTM	A chemical modification of an amino acid that alters its properties, such as phosphorylation of serine, threonine or tyrosine.
Profile	PSSM, PSWM, PWM, HMM	An extended representation of a sequence where each position accounts for variability between elements of the alphabet. Also known as a Position (Specific) Scoring/Weight Matrix (PSSM/PSWM/PWM). For the purposes of this review, hidden Markov models are also referred to under the "profile" umbrella.
Protein-Protein Interaction	PPI	A physical interaction between two proteins.
Regular Expression	Regex, PROSITE pattern	A common programming notation for string (text) patterns. For the purposes of this review, variants on the standard regular expression notation are included under the "regex" umbrella.
Short Linear Motif	SLiM, Linear Motif, LM, Minimotif	A short (typically <15 aa) linear stretch of protein sequence with specific residues important for function. Within this review, "motif" refers to a SLiM unless otherwise specified.
Support	UP Support	The number of different proteins that contain a given SLiM. "UP" indicates that this is the number of *unrelated* proteins.
Wildcard		A position in a SLiM that can be any amino acid.

(PSSM/PSWM/PWM) and hidden Markov models (HMM), expand on the simplicity of regular expressions by adding a quantitative dimension.

1.2.1 Regex Representations of SLiMs

The main elements of regular expressions are provided in Table 3. Evidence for SLiMs and the contribution of individual residues to their function comes from a variety of sources but is essentially either positive (specific residues are critical for function) or negative (presence of specific residues ablates function). At the two extremes, presence of a single specific amino acid side chain can be necessary for function or sufficient to block binding. Where a SLiM

Table 3
SLiM regex elements

Regex	PROSITE	MnM	SLiMSuite	Description
A	-A-	A	A	A single fixed amino acid, A using standard IUPAC letters.
[ILV]	-[ILV]-	[ILV]	[ILV]	Either I, L or V. Can have any number of possible amino acids.
[^P] or [^DE]	-{P}- or -{DE}-		[^P] or [^DE]	Exclude one or more amino acids.
.	-x-	X	X or .	Wildcard. Any amino acid.
.{n}	-x(n)-		.{n} or X{n}	A repeat of *n* wildcard positions.
.{m,n}	-x(m,n)-		.{m,n} or X{m,n}	A repeat of at least *m* and at most *n* wildcard positions. (*m* can be zero.)
^	<	<	^	N-terminus of protein.
$	>	>	$	C-terminus of protein.
(p1|p2)			(p1|p2)	Either regex pattern p1 or p2.
r{n}	r(m)		r{n}	*n* repetitions of *r*, where *r* is one of the above regex elements.
r{m,n}	r(m,n)		r{m,n}	At least *m* and up to *n* repetitions of *r*, where *r* is one of the above regex elements.
			<r:n:m>	At least *m* of a stretch of *n* residues must match *r*, where *r* is one of the above regex elements (single amino acid, ambiguity or exclusion list).
			<r:n:m:b>	Exactly *m* of a stretch of *n* residues must match *r* and the rest must match *b*, where *r* and *b* are each one of the above regex elements.
			(ABC)	A, B and C in any order.

forms a helical structure upon binding, for example, the presence of a proline may disrupt this. In between these extremes, a number of different amino acids may be tolerated at a given position and still give rise to a functional SLiM instance. Such positions are referred to as "degenerate" or "ambiguous" and consist of sets of amino acids with certain common properties, such as positive charge. Fully degenerate positions that can tolerate any amino acid are referred to as "wildcards" and typically represented with the symbols "." or "X". Sometimes these can also be referred to as "gaps" in a motif, which can be confusing to the unwary and have nothing to do with gaps (insertion/deletion events) in sequence alignments. When regular expressions are derived from sequence alignments, indels in the latter are generally represented by flexible-length wildcards (Table 3).

Regular expressions are purely qualitative, which makes them easy to model and amenable to fairly simple but effective statistics [17]. Another advantage of regular expressions is that they already form part of numerous programming and scripting languages and can therefore be used for simple computational SLiM discovery with minimal overheads. It should be noted that the PROSITE [18] and MnM [10] SLiM repositories have their own variants of regex notation (Table 3), which may need converting to standard regex patterns prior to analysis with other tools or servers. SLiMSuite tools can make this conversion if required. Some tools have expanded the standard regex patterns, as discussed below.

1.2.2 Profile Representations of SLiMs

Most profile-based methods represent SLiM-like sequence signatures as matrices that are derived from input data containing a set of sequences assumed to carry the motif of choice. These can be user-specified after careful inspection of interesting data or extracted from larger datasets using computational methods. Profiles are typically derived from a frequency table of $20 \times N$ fields (with N being the length of the motif), which is computed from the site-specific amino acid counts and normalized by the number of input sequences and inherent biases in amino acid composition. The latter is usually taken from an empirical background distribution or collected from randomized sequences. Building a profile from a restricted set of known sequences can omit valid occurrences of amino acids at positions where they were not observed. To avoid this it is customary to use "pseudocount" observations, which are added into the frequency table even though they were not actually observed. Since the contribution of pseudocounts is small and continues to diminish as more observed data is added, they will have little relevance to the final profile but are crucial mathematically to avoid the issues of null values in log-odds profile representations. The resulting profile should be an overrepresented sequence signature as observed in the data, from which a putative motif could be extracted [19]. Such a profile can be considered as a special, limited case of the profile hidden Markov model (pHMM) [20, 21]. The added versatility of pHMM comes from their capacity to not only assign different frequencies to residues but also to allow for insertions and deletions of variable length between sites.

On face value, a profile is superior to a regular expression in many ways because the frequency data allows quantitative scoring of a motif instance. Whereas a regular expression might have [ILV] for a position, a profile could encode the information that 90 % of instances have an isoleucine (I) and only a minority have leucine (L) or valine (V) and weight observations accordingly. The drawback is the requirement for sufficient data to make the profile accurate. SLiM instances are generally few in number and so there is a big danger of over-fitting the profile model, especially given that there are 20 possible states for each position in the motif. Rather than modelling the true constraints of the motif, profiles could simply be representing any early bias in discoveries. For this reason, profiles tend to be popular for DNA motifs (where the number of instances is often high and there are few possible states per position) but are of much more limited use for SLiMs and other alignment-free protein motifs. Exceptions are posttranslational modifications (PTM), such as phosphorylation, which often have many occurrences and recognition motifs based on large screenings of peptide libraries (well-exploited in methods such as Scansite [22]). Where sufficient data exists, profiles can be very powerful because of their ability to quantitatively assess deviations for core SLiM consensus definitions. This can help when identifying previously unseen variants of known motifs and could prove essential to effectively mine large data in the search for novel SLiM instances.

1.2.3 Limitations with Current Motif Definition Schema

The common SLiM formats do have some limitations in the nature of information that they can encode. There is currently no good way to represent interdependencies between sites, for example, where the constraints on one position are determined by the amino acid at another. For profiles, context-sensitive HMM [23] may help to model non-contiguous relationships but are yet to be widely applied in bioinformatics. For regex motifs, some effort has been made in this direction with the 3of5 webserver [24], which recognizes "*n* of *m*" stretches where *n* residues in a window of length *m* are of a given type. This was extended further by PRESTO (Table 1) and its successor in the SLiMSuite package, SLiMProb (formerly SLiMSearch 1.x [25]), to allow more complex either/or stretches in the form "<*r*:*n*:*m*:*b*>", where *r* and *b* can be any single or ambiguous regex elements (Table 3) of which *n* residues in a window of *m* positions must match *r* and the remainder must match *b*. If *b* is a wildcard then a simpler "<*r*:*n*:*m*>" notation can be used, which corresponds to the original "*n* of *m*" pattern element. This notation allows very efficient encoding of complex regex patterns, although these are actually exploded by SLiMProb into different sub-variants for searching. For more complex scenarios, multiple versions of a motif are defined, such as the Class I and Class II SH3 domain motifs in ELM, [RKY]xxPxxP and PxxPx[KR], respectively [9]. Current SLiM definitions also do not encode the secondary/tertiary structural constraints [26], even if some of the SLiM databases do store and utilize such information for specific motif entries, as described in later sections.

1.3 SLiM Evolution

There are two key principles underlying the evolution of SLiMs in protein sequences: conservation of individual SLiM occurrences (divergent evolution) and independent evolution of SLiM occurrences in unrelated proteins (convergent evolution) [1]. The functional constraints of SLiMs mean that they are subject to purifying selection and will generally show a higher level of conservation than the surrounding residues in disordered regions [27]. The evolutionary plasticity of SLiMs is generally higher than residues that are both functionally and structurally constrained, with single point mutations often sufficient to destroy a motif occurrence or even create a functional SLiM from previously inactive protein sequence. Such plasticity may be harnessed by positive selection to rapidly rewire PPI networks, particularly considering that the low affinity nature of SLiM-mediated DMI probably confer an extra tolerance of SLiM gains and losses in the network [12].

There is no one-size-fits-all solution to SLiM discovery and one must carefully consider the nature of the data before selecting the evolutionary models that should be applied. Where occurrences are likely to be functionally relevant and there is reason to suspect that this function would be found in ancestors, e.g., it encodes a function seen across all mammals/vertebrates, it makes sense to look for signals of evolutionary conservation on a background of divergence. If, on the other hand, a SLiM occurrence is speculated to be new (in evolutionary terms) or even nonphysiological (obtained from experiments such as peptide library screens or yeast two-hybrid data) then evolutionary conservation will be misleading at best and counter-productive at worst. The nature and distribution of the SLiM occurrences must be considered before invoking a model of convergent evolution. Phage display is essentially convergent evolution in the laboratory, whilst random peptide libraries have the sequence independence of convergence even if there has been no evolution per se. If, on the other hand, all known occurrences of a SLiM come from the same protein family then conserved function is the most parsimonious explanation and it makes little sense to model convergence. The exception, of course, is where additional evidence points to multiple independent origins of SLiM function.

1.4 SLiMs and Protein Structure

The majority of SLiMs occur in intrinsically disordered regions (IDR) of proteins, at least in their unbound state [1]. The reduced structural constraints of IDR result in reduced evolutionary constraints and mean that they are generally free to evolve at a faster rate at the sequence level [28], even if they generally conserve their ordered/disordered protein segments [29]. This, in turn, contributes to the previously mentioned evolutionary plasticity of SLiMs, in addition to conferring a degree of structural flexibility on SLiMs that includes potential disorder-to-order transitions linked to protein binding [30]. Indeed, certain SLiMs are known to undergo conformational rearrangements of this type [8, 31], although this is unlikely to be the case for all SLiMs. Following molecular dynamics simulation, Cino et al. have recently proposed that SLiMs tend to adopt conformations typical of their bound state even in the free state [32]. Under this model, "pre-equilibrium" structured SLiM conformations are stabilized later by the interaction, as opposed to an "induced fit" model where binding itself triggers the conformational change. Either way, whilst flanking regions of SLiMs tend to match the composition of IDR [30], the key positions in SLiMs themselves are enriched for hydrophobic and aromatic amino acids more typical of structured regions [1]. Indeed, many SLiMs can be thought of as regions of disorder with a propensity towards order [30]. This flanking disorder may itself be under positive selection to confer protection against peptide aggregation around SLiMs [33]. How much of the enrichment of SLiMs in IDR versus globular regions is due to structural constraints for SLiM-mediated binding, and how much is simply due to the increased evolutionary plasticity of IDR increasing the chance of SLiMs evolving convergently, is yet to be established.

SLiMs include PTM sites, some of which occur on the (structured) surface of globular domains. There are also extracellular SLiMs, which occur in proteins with less intrinsic disorder than intracellular proteins [1]. As a result, approx. 15 % of all known SLiM instances are actually on globular domains. These regions can present extra challenges for SLiM discovery, as they will also contain a number of structural motifs that are constrained in three-dimensional space. Whilst not necessarily linear, many structural motifs will include linear stretches that could be erroneously identified by SLiM predictors. Methods for predicting structural motifs directly do exist (e.g., SiteBinder [34]), but these are not considered in this review.

1.4.1 Protein Isoforms and SLiMs

SLiMs are undoubtedly responsible for some of the functional diversity imparted on protein sequences via alternative splicing/promoter use [35–37]. Alternative translation initiation sites can also give rise to different protein products [38, 39] and are likely to similarly alter the SLiM complement of proteins, particularly in terms of N-terminal subcellular targeting motifs. To date, however, most resources for both PPI and SLiM prediction deal predominantly with "canonical" protein sequences and thus protein isoforms are not further considered in this review. All of the approaches described can potentially be applied to protein isoforms and this flexibility represents one of the benefits of tools that permit analysis of bespoke protein sequences rather than relying on, for example, Uniprot [40] data. It should also be noted that methods such as SLiMFinder [41] that correct for evolutionary relationships within input sequences should also be able to deal with multiple isoforms for each protein, although this has not been formally tested. Note also that GOPHER [42], which is supplied with SLiMSuite, can be used to generate alignments of orthologous splice variants from appropriate source data, such as Ensembl proteomes [43].

1.4.2 SLiMs, MoREs and MoRFs

SLiMs are not the only binding features present in IDR. Regions within IDR that mediate PPI via a disorder-to-order transition upon binding have also been labelled Molecular Recognition Elements (MoREs) [44] (if reasonably short and helical) or Molecular Recognition Features (MoRFs) [45, 46]. There is not a clear delineation between the concepts of SLiMs and MoREs/MoRFs. Some classes of SLiMs probably represent a subset of MoRFs that are short and have specific residues involved in the interaction. Other SLiMs are too short to count as MoRFs (defined as 10–70 aa in length) and/or do not undergo the stipulated structural transition. SLiMs and MoRFs are therefore best considered as complementary and overlapping sets of molecular features. IDR-mediated PPI may include either, both or indeed neither element [6].

1.5 Definition and Databases of Known SLiMs

This review is predominantly concerned with the task of predicting SLiMs from one or more protein sequences. Before examining the primary methods for doing so, it is useful to briefly consider where our current SLiM definitions come from as well as the key databases for storing them. SLiMs are notoriously difficult to define and one must always entertain the notion that definitions found in SLiM databases are incomplete and/or biased by the nature of their discovery. Most known SLiMs were experimentally discovered, although precisely defining the motif often involves bioinformatics, such as a sequence alignment, and manual decisions regarding what comprises the important and/or conserved residues. Often, motifs are simplified to a “canonical” core but also have “non-canonical” instances that deviate from the main definition. This can create some confusion for SLiM rediscovery as it is not always clear what definition(s) of a motif to use. SLiMs are affected by their immediate context, with flanking residues that do not seem to increase affinity directly but are crucial to the specificity of binding [47]. It is therefore highly likely that the flanking sequence could add binding constraints that would render certain residues superfluous. SLiM predictions do not normally tolerate mismatches because the SLiMs themselves already have very low information content and a high probability of occurring by chance.

In situations where non-canonical occurrences are common search tools that incorporate mismatches (e.g., PRESTO) might be required.

When considering the experimental evidence for SLiMs, the nature of the protein–peptide interaction and whether it provides biophysical or biological support is important. In other words, is the experiment providing evidence of what *could* bind or what *does* bind? High-throughput experiments, including screening peptide libraries and similar technologies such as phage display can potentially define binding motifs without any known PPI. This approach can have advantages, in that it can potentially define motifs for "singleton" interactions (e.g., those with only a single occurrence in nature) and can also generate the high numbers of sequence variants required for building profiles, as exemplified for PDZ and SH3 domains [48] and for PTM by Scansite [22]. The high number of variants is also good for identifying amino acids that are not tolerated in particular positions, which otherwise tends to require careful mutation studies. It should be remembered, however, that such SLiMs are not always physiological: peptide-based techniques will be biased towards sequences that have the strongest affinity, whilst SLiMs in nature often have a lower affinity than possible in order to maintain the correct signalling dynamics [1, 8, 15]. This lack of physiological relevance is not necessarily an issue and permits the exploitation of data that might otherwise be ignored. For example, Liu et al. have found evidence from yeast two-hybrid experiments that out-of-frame constructs, which code for short peptides without homology to known proteins and are typically discarded as false positives, may contain novel SLiMs that can be identified computationally [49].

There are now a number of public repositories that are largely or wholly dedicated to collating and curating SLiMs from the literature. These are an excellent source of known motifs and motif instances, which can be used either to interrogate a protein of interest or to assess a potentially new SLiM discovery. An overview of the four main SLiM databases is given below. In addition, a number of targeted motif databases exist for specific classes of SLiM, particularly PTM [50, 51].

1.5.1 PROSITE

PROSITE was one of the earliest collections of linear motif definitions for both SLiMs and longer globular domains [18] although it has largely been superseded by ELM [9] and MnM [10] as a repository for SLiMs. PROSITE motif notation is similar to standard regular expression notation but has some important differences (Table 3). Its regex domain definitions provide a potential source for identifying putative SLiMs that are actually structural motifs or parts of larger regions of homology. For domain searches themselves, it is more usual to use the sequence profiles in PROSITE [52, 53] or HMMs (e.g., SMART [54] and Pfam [55]).

1.5.2 ELM

The Eukaryotic Linear Motif (ELM) database is now over 10 years old [9] and the number of annotated ELMs (as of Jan 2014) has increased to nearly 200 classes and over 2,400 instances in six categories (as denoted by their prefix):

- CLV: Proteolytic cleavage. Sites of posttranslational enzymatic cleavage.
- DOC: Docking. These recruit a modifying enzyme but are not targeted by the active site.
- DEG: Degron. Part of the proteasomal degradation pathway, directing protein polyubiquitination.
- LIG: General ligand binding. Mediating PPI is the primary/sole known function.
- MOD: Posttranslational modification sites, e.g., phosphorylation. (Note that proteolytic cleavage has its own CLV class).
- TRG: Subcellular targeting. Recognized by machinery that directs the parent protein to appropriate cellular localization.

Note that the DOC and DEG categories are recent additions and many studies will have these motifs classed as LIG under the previous classification. The remaining LIG category can best be thought of as SLiMs for which the main, or possibly only, known function is to mediate a PPI. Arguably, all ELMs are protein ligands but it can be useful to consider distinct subsets in case they have different biases in attributes and behavior. Indeed, a recent review using the older four-category classification highlighted some differences between ligands, modifications, and targeting sites [1]. Future releases may extend this classification further.

In addition to the database, ELM hosts a motif search server that includes built-in filters based on evolutionary conservation [56] and structural considerations [57], which are explored in more detail in later sections. Other resources at ELM include the iELM server for exploring SLiM interactions [58], the Phospho.ELM database of experimentally verified phosphorylation sites [59], the switches.ELM "compendium of conditional regulatory interaction interfaces" [13, 60] and a curated set of eukaryotic SLiMs that are the target of molecular mimicry by viral proteins [15]. The ELM conservation scorer is also available at the site to run on user-supplied proteins or alignments. Although MnM has more instances than ELM (even with the 43,000 Phospho.ELM sites), the quality of the curation and availability of the data make ELM the leading SLiM repository.

1.5.3 Minimotif Miner (MnM)

Minimotif Miner (MnM) probably has the largest collection of known SLiMs from the literature, with over 295,000 instances and 880 consensus sequences in MnM 3.0, of which the vast majority are PTM [10]. Some of these are redundant, and so the real number is likely to be somewhat smaller. Figure 3 of the MnM 3.0 paper, for example, lists both Rx[KR]R and Rx[RK]R as furin proteolysis motifs. MnM is enriched in mammalian motifs but not restricted to eukaryotes by design, with some entries found in bacteria. As with ELM, these are available for searching against an input sequence using an online search tool (*see* next section). Unfortunately, unlike ELM, MnM have not made their SLiM collection available to download and interrogate outside of their webserver, which limits the utility of the service.

1.5.4 Scansite

Many SLiMs are recognition sites for reading, writing or erasing PTMs. Phosphorylation is particularly widespread in signalling systems [61]. Scansite is a leading database for phosphopeptide motifs and the premier profile-based SLiM database and search tool [22]. Scansite3 is its latest version and has profile models for 70 mammalian and 54 yeast protein kinases and phosphopeptide binding domains (e.g., 14–3–3, SH2, SH3, PDZ). The majority of the data in Scansite were generated using "oriented peptide libraries", which fix a central (possibly phosphorylated) serine, threonine or tyrosine residue, and generate random libraries of flanking sequences that are incubated with the domain of interest [62]. Subsequent Edman sequencing of phosphorylated/bound peptides generates the amino acid frequency distribution at each position, which is then converted into a sequence recognition profile. These data are excellent at identifying the optimal (e.g., highest affinity) binding profile for a given phosphopeptide domain. It should be noted, however, that biologically relevant SLiM occurrences are not necessarily optimized for maximum binding affinity [1, 8, 15] and may therefore show a profile different from those generated by a peptide library screen.

1.6 Databases of Predicted SLiMs

There are currently no databases collecting and/or annotating predicted SLiMs but several large-scale SLiM predictions have their data available as supplementary data and/or online (Table 4). Interpreting SLiM predictions is largely a matter of placing them in context. Results from an interactome-wide de novo SLiM prediction in humans [63], for example, have been made available as a series of linked web pages called SLiMdb. This enables predictions to be grouped and studied by SLiM, hub (i.e., proposed PPI partner), parent protein, and GO classification [64]. Entries link out to data in external resources including Ensembl [43], Uniprot [40], GO, OMIM [65], HPRD [66], and Genecards [67]. Further context can be provided by searching the motifs against the human proteome using SLiMSearch2 [68]. Predicted SLiMs have also been compared to each other and/or to databases of known motifs using CompariMotif [69], which is available both as a webserver and a standalone program. This enables clusters of similar motifs to be identified and explored. In future, it is planned to extend

Table 4
Large-scale de novo SLiM Discovery analyses

Method	Data	Source	Species	Data available?	Predictions available?	Reference
FIRE-pro	GO, PPI, sub-cellular localization, half-life	Online databases and curated bibliography	Yeast	Y (with formatted data for other species)	Y	[124]
SLiMFinder	PPI	Online databases	Human	Y	Y	[63]
LMD (DILIMOT)	PPI	Yeast two-hybrid, online databases and curated bibliography	Human, fly, nematode, yeast	N (retrievable from original authors)	Y	[71]
D-STAR	PPI with SH3 domains and in TGFβ signalling pathway	Online databases and curated bibliography	Yeast	N (retrievable from original authors)	Y (partial)	[121]
motif-x	Phosphopeptides	Immunoaffinity and SCX Chromatography	Human	N (retrievable from original authors)	Y (from publication)	[135]
motif-x and scan-x	PPI	Online databases	Human, mouse, fly, yeast	N (retrievable from original authors)	Y	[136]
motif-x	Phosphopeptides	LC/MS-MS	Fly, mouse, yeast	Y	Y	[137–139]
MeMotif	Transmembrane proteins	Online databases	All	N (retrievable from source)	Y	[70]

SLiMdb to improve data querying and include data from other SLiM prediction studies. Another example of a resource in which motif predictions have been made available for interactive exploration is the MeMotif database of consensus linear motifs from alpha-helical transmembrane protein structures [70].

1.7 DNA and Protein Motif Search Tools

Most motif prediction tools developed for DNA or protein sequence motifs can be adapted to the other biopolymer by simply changing the alphabet. There are important differences between DNA and protein sequences, however, and these should not be ignored or overlooked. DNA is simple in comparison, with only four possible base states (ignoring methylation) and DNA sequences analyzed tend to be relatively long, from hundreds to millions of bases. This enables a much more accurate modelling of the background sequence space, even at the di- or tri-nucleotide level; if protein-coding regions are present in the DNA, amino acid and codon usage bias might result in tri-nucleotide sequence biases. DNA motifs also tend to have much higher support in search datasets. In contrast, protein sequences are much shorter (the average unmasked human protein being approx. 500 aa in length) and there are 20 amino acids (excluding PTM), which means that di-amino acid frequencies can rarely be accurately estimated. Protein motifs also tend to have fewer occurrences. As a result, sophisticated methods that can work well for DNA motif prediction are usually either inappropriate or impractical for protein motif applications. For this reason, this review concentrates on tools that have been explicitly designed and/or benchmarked for de novo SLiM discovery. Exceptions include algorithms designed to identify motifs from individual proteins based on alignments of orthologues, for which it is possible to assemble quite large datasets, and the Scansite peptide library approach described above [22, 62].

1.8 SLiM Discovery Benchmarking

A challenging and sometimes overlooked aspect of both the development and appropriate application of SLiM discovery tools is robust benchmarking. New methods require adequate benchmarking data to ascertain their utility and whether they offer an improvement over existing methods. The latter comparison can be particularly hard if the most similar methods have themselves been inadequately benchmarked. The latest releases of SLiMSuite include SLiMBench (unpublished), a tool for generating SLiM discovery benchmarking datasets and assessing performance. SLiMBench and some model benchmark datasets will be made available at the SLiMSuite website. In the meantime, here are some considerations for the benchmarking of SLiM discovery tools:

- *Scale.* Whilst they can be useful exemplars for specific method features, individual observations do not constitute benchmarking and are easily subject to performance bias, whether deliberate or accidental. Regrettably, a number of the less specialized tools are benchmarked quite well on DNA data but neglect protein applications with a limited number of poorly conceived test datasets. The restricted number of known SLiMs does present a problem and previous methods have been somewhat limited in terms of benchmarking on real data (*see* for example DILIMOT [71] and SLiMFinder [41]) but, at the very least, the ELM database should be used [9]. Simulated data is also useful for getting the numbers up, subject to the considerations below.

- *Bias.* Benchmarking data should ideally be unbiased but, at the very least, its biases must be clear. It is OK to include a benchmark that is biased towards the particular model being tested but this will only tell you whether the algorithm is working computationally, not whether it is useful biologically. Benchmarks should also include data that does not make the same assumptions as the new model being tested. Particular attention should be paid to dataset size and signal:noise ratios in the data as methods generally perform better when these are both large, which is not always realistic.
- *Realism.* Regardless of benchmarking performance on simulated data, there is always the possibility—indeed, likelihood—that real data will have additional biases. Checking performance against real data is therefore crucial. It is often hard to get the same numbers as with simulations and the assessments are often less robust as a result but this cannot really be helped. (Re-benchmarking later is always an option.). The important thing is that benchmarking is not solely on simulated data.
- *Accuracy versus efficiency.* Although computational efficiency is important, accuracy of methods is more important. Whilst a slow method can often be overcome by careful parameter selection and/or finding a faster/bigger computer, rapid results of unknown accuracy are of limited use.
- *False Positive Rates.* Often, methods are only benchmarked in terms of recovery of true positive motifs. A frequent approach is to rank predictions and then demonstrate that the known motif is returned among the top-ranked motifs and/or most of the top-ranked motifs are true positives. The problem with this is that all such test datasets have motifs to be found. In real biological scenarios, it is often not known whether (a) there is a real motif in the data to be found and/or (b) if so, whether there is actually enough signal for said motif to occur more than by chance. For de novo discovery, it is imperative that methods are also benchmarked on datasets that have no real/planted motifs and thus all predictions are false positives. Simulated data is particularly useful for modelling these.

To carry appropriate SLiM predictions forward to laboratory validation, an estimate of the likelihood that a given returned motif is a false positive is essential.

- *Application-focused*. Predictive bioinformatics tools frequently make use of Receiver Operating Characteristic (ROC) curves, particularly for classification problems such as predicting known motifs. ROC curves, which plot true positive rates (TPR, the proportion of positives that are correctly predicted) against false positive rates (FPR, the proportion of negatives that are wrongly predicted) for different thresholds, can be useful exploratory tools but they leave a lot to be desired when it comes to biological benchmarking. Usually, one is operating in a specific part of the ROC space—typically, either minimizing FPR or maximizing TPR. Tools may excel in one area but do poorly in another and this is not captured adequately by "Area Under the (ROC) Curve" (AUC) statistics. When selecting a SLiM discovery tool, it is best to choose one and/or select parameter settings that perform appropriately for the desired application. SLiMFinder, for example, is extremely stringent, making it ideal where false positives are to be avoided [41]. Where false predictions are not an issue, however, the SLiMChance ([17, 41]) significance threshold of SLiMFinder can be relaxed to increase sensitivity.
- *Comparative*. Whenever possible, new methods should be compared to existing methods where they are available (Table 5).

True positives can be identified by comparing predicted SLiMs to databases of known motifs, either manually or using CompariMotif [69]. Although motif comparisons are scored and ranked by CompariMotif, we are not currently aware of a statistical framework for these comparisons nor is there a well-modelled threshold for assigning a match between two motifs. In order to consider a SLiM to be a true positive, SLiMBench uses CompariMotif criteria that matches must: [1] have 2+ positions match; [2] have a normalized information content of at least 1.5 (approx. equivalent to 1 fixed position and one mildly degenerate position, *see* [69]); [3] match at least half of the smallest motif. Although strict application of these criteria will misclassify some motif matches, agreement with our manual classification was good (data not shown) and it has the advantage of being consistent and unsupervised, which is clearly beneficial for comparative benchmarking. For the moment, however, motif matches are largely a matter of individual discretion; caution and discretion should be employed when interpreting claims of "true positive" motif predictions, especially in the absence of data being made available.

Table 5 Computational SLiM Discovery methods

Tool	Description	Ref.	Availability[a]	Prediction[b]	Notation[c]	Input[d]	Scoring[e]
AMS	AutoMotif Server. Machine Learning predictions of PTM	[101]	D	K	O	M	S
ELM	Eukaryotic Linear Motif server. Regex searches of known SLiMs with numerous contextual filters	[9]	W	K	R	S	C D O
iELM	Interactions of Eukaryotic Linear Motif. Predict new instances of known ELM motifs from PPI data	[58]	W	K	R	S M	D O
iSPOT	Infer Sequence Prediction Of Target. Prediction of PDZ, SH3 and WW binding sequences from structural data	[91]	W	K	P	M P	O
MnM	Minimotif Miner. Regex searches of curated literature motifs with numerous contextual filters	[10]	W	K	R	S	C O S
ScanProsite	Perform Regex and Profiles searches of PROSITE patterns or user-defined motifs against user proteins or public databases	[76]	W	K U	R P	S M P	O
Scansite	Profile-based searches of known phophoSLiMs against user sequences. Searches of user-defined regex and profile motifs against public databases	[22]	W	K U	R P	S P	O P
3of5	3of5 regex search tool. Simple protein searches with expanded regex notation	[24]	W	U	R	M	
ANCHOR	Identifies regions with propensity for order within IDR. Can map user-defined regex onto disorder profiles	[125]	W	U D	R	S	D S
FIMO	Find Individual Motif Occurrences (MEME Suite). Search MEME profiles against user proteins or public databases	[73]	W D	U	P	M P	C S
GLAM2SCAN	Scanning with Gapped Motifs (MEME Suite). Search GLAM2 profiles against user proteins or public databases	[74]	W D	U	P	M P	C
MAST	Motif Alignment and Search Tool (MEME Suite). Search with multiple profile motifs in combination for proteins with high combined scores	[75]	W D	U	P	M P	C S

(continued)

Table 5
(continued)

Tool	Description	Ref.	Availability[a]	Prediction[b]	Notation[c]	Input[d]	Scoring[e]
PRESTO	Protein Regular Expression Search Tool (SLiMSuite). Regex search tool of user-defined SLiMs against local protein data with expanded regex notation and tolerance of mismatches	[140]	D	U	R	M	C D O
SLiMProb	Short Linear Motif Probability (SLiMSuite). Formerly SLiMSearch 1.x. Regex search tool of user-defined SLiMs against local protein data with expanded regex notation and numerous contextual masking options. Returns significantly over- and under-representation statistics controlling for homology	[25]	W D	U	R	M	H C D O S
SLiMSearch2	Short Linear Motif Search (SLiMSuite). Proteome screen of regex with contextual filters	[68]	W D	U	R	P	C D O
SLiMScape	Short Linear Motif analysis plugin for Cytoscape (SLiMSuite). Can run SLiMProb or SLiMFinder on proteins selected within Cytoscape	[99]	D	U D	R	M	H C D O S
D-MIST	Domain-Motif Interactions from Structural Topology. Machine Learning predictions of DMI from PDB based on structural context	[131]	D	D	P	P	D O
D-MOTIF	LDMS CMM tool. Identifies correlated motifs in PPI data	[121]	D	D	R	P	
D-STAR	LDMS CMM tool. Identifies correlated motifs in PPI data	[121]	D	D	R	P	
DILIMOT	DIscovery of LInear MOTifs. Formerly LMD. Models convergent evolution/over-representation of TEIRESIAS regex motifs with evolutionary and structural filters	[71, 113]	W	D	R	M	H C D
FIRE-pro	Finding Informative Regulatory Elements in proteins. LDMS CMM tool using Mutual Information to identify motifs that correlate with biological features	[124]	W D	D	R	P	O S

GLAM2	Gapped Local Alignment of Motifs (MEME Suite). Profile-based de novo prediction of over-represented patterns using Gibbs sampling and simulated annealing	[74]	W D	D	P	M	
MEME	Multiple Em for Motif Elicitation (MEME Suite). Profile-based *de novo* prediction of over-represented patterns using expectation maximisation	[114]	W D	D	P	M	
MFSPSSMpred	Masked, Filtered and Smoothed Position-Specific Scoring Matrix-based Predictor. Identifies short regions with propensity for order within IDR based on sequence features and evolutionary conservation	[106]	W D	D	O	S	C D O
MoRFpred	MoRF predictor. Identifies regions with propensity for order within IDR	[128]	W	D	O	S	D
motif-x	Generates fixed position motif from alignment peptides based on over-representation versus background amino acid frequencies	[107]	W	D	R	M P	
MotifCluster	LDMS CMM tool. Identifies correlated motifs in PPI data	[122]	D	D	R	P	O S
MOTIPS	MOTIf analysis Pipeline. De novo profile prediction based on over-representation in short aligned peptides combined with domain-based PPI data	[134]	W D	D	P	M P	C D
NestedMICA	Nested Motif Independent Component Analysis. Identification of enriched motifs versus background reference proteins	[117, 118]	D	D	P	M	
PepSite	Predicts possible DMI from peptides and structural data	[92]	W D	D	R	S	O S
qPMS7	Over-representation of LDMS patterns without correction for homology	[119, 120]	W D	D	R	M	
Pratt	Over-represented regex motif prediction without correction for homology	[110]	W	D	R	M	

(continued)

Table 5 (continued)

Tool	Description	Ref.	Availability[a]	Prediction[b]	Notation[c]	Input[d]	Scoring[e]
QSLiMFinder	Query SLiMFinder (SLiMSuite). Query-based de novo regex SLiM prediction modelling convergent evolution with correction for homology, numerous masking options and statistical support	[41]	W D	D	R	M	H C D O S
SLIDER	LDMS CMM tool. Identifies correlated motifs in PPI data by mapping motifs onto PPI interfaces using structural data	[123]	D	D	R	P	D O
SLiMDisc	Short Linear Motif Discovery (SLiMSuite). Regex de novo SLiM prediction modelling convergent evolution with correction for homology and numerous masking options	[42, 77]	W D	D	R	M	H C D O
SLiMFinder	Short Linear Motif Finder (SLiMSuite). Regex de novo SLiM prediction modelling convergent evolution with correction for homology, numerous masking options and statistical support	[41, 100]	W D	D	R	M	H C D O S
SLiMMaker	Short Linear Motif Maker (SLiMSuite). Simple regex consensus generator from aligned peptide sequences	[140]	W D	D	R	M	
SLiMPred	Short Linear Motif Predictor (SLiMSuite). Artificial Neural Network predictor of SLiMs from sequence features	[129]	W	D	O	S	C D
SLiMPrints	Short Linear Motif fingerprints (SLiMSuite). Prediction of SLiM conservation fingerprints using statistical modelling of RLC	[97]	W	D	R	S	C D S
TEIRESIAS	Simple but efficient text pattern search tool	[112]	D	D	R	M	

[a] Availability of software: *W* Webserver, *D* Download

[b] Prediction type: *K* Known motifs, *U* User-defined, *D* De novo

[c] SLiM notation used: *R* Regex, *P* Profile, *O* Other

[d] Input data for searching: *S* Single Protein, *M* Multiple Proteins (methods accepting multiple proteins can usually be scaled for single proteins or proteomes), *P* Proteome/ Database

[e] Scoring and filtering options: *H* Homology correction, *C* SLiM conservation, *D* Structure/disorder filtering/masking, *O* Other filters/scores, *S* Significance estimate

2 Computational Prediction of Known SLiMs

Computational techniques for profile and regex searches are well established and thus finding predefined patterns in sequences is a computationally trivial exercise. Due to their short length, SLiMs are very likely to be found in proteins of typical sizes; the difficult job is distinguishing genuine functional instances of a SLiM from random background occurrences. Choosing the right search tool for known SLiMs is therefore largely governed by what is known about the motifs in question.

Many of the repositories of known motifs also include tools for searching those motifs against a given protein, including the Eukaryotic Linear Motif (ELM) resource [9], Minimotif Miner (MnM) [10] and Scansite [22]. There also exist a number of bespoke tools for searching user-defined motifs against protein datasets (Tables 1 and 5) and no doubt more will be added in the future. ELM and MnM have all-or-nothing matches based on their regular expressions, which are then rated and/or filtered according to contextual information to help the user discriminate true positives from false positives. SLiMProb [25] and SLiMSearch2 [68] allow similar searches for sets of proteins and whole proteomes, respectively, using user-defined regular expressions. Both also provide contextual scoring/ranking options and output that permits users to visually explore predicted instances.

Scansite [22] harnesses the power of probabilistic profile models of known cell signalling interaction motifs to predict new instances in user-defined sequences or various public protein databases. The Scansite3 server can also make SLiM predictions with user-specified matrices of binding affinities per site, enabling users to easily search with their own profiles. Additional flexibility in motif definition is introduced by allowing the specification of an approximate consensus sequence of the motif, which is then used to automatically construct a matrix with similar characteristics. Scansite3 also features an option for searching user-defined peptides or regular expressions against a selection of protein databases. MEME Suite [72] offers a set of scanning tools to allow searching sequence databases with the profile motifs, such as those identified de novo by other tools in the suite. Ungapped motifs found by MEME can be used as input for FIMO [73] to find all motif occurrences in a public protein database, ranked by significance according to their Benjamini–Hochberg corrected *p*-values. GLAM2SCAN offers the same functionality for input gapped alignments provided by GLAM2 [74]. A slightly different approach is taken by MAST [75] as it considers the full set of input motifs as a whole. It first determines the best scores for all matches between pairs of motifs and proteins in the database and then combines these into overall scores between the complete set of motifs and each protein. The *E*-values calculated by MAST are used to filter out random hits (with a user-defined threshold) and rank the remaining significant proteins. Since MAST results provide a single score for each protein in the database, and information from multiple motifs can be provided as input, the program could be useful to retrieve proteins where different motifs co-occur.

PROSITE patterns can be searched online using Scanprosite [76]. Scanprosite allows proteins to be scanned for PROSITE patterns, or the user can define patterns to be searched against public sequence databases or user-defined protein datasets. Because it does not have any of the filtering tools advised for SLiM discovery (discussed below), Scanprosite is not recommended for SLiM prediction. Users should also note that the Scanprosite default is to "exclude motifs with a high probability of occurrence from the scan," which includes many of the SLiMs in the database. SLiMProb can perform local searches using PROSITE patterns in place of standard regex notation. Likewise, CompariMotif [69] can be used to compare regex motifs with PROSITE patterns.

In general, there are two assessments of SLiM predictions that a user wants to perform: assessing individual predicted occurrences, or assessing enrichment of a dataset for predicted occurrences. These are explored in more detail below.

2.1 Assessing and Ranking Individual SLiM Occurrences

SLiM discovery methods are notorious for over-prediction. To combat this, there are a number of possible considerations that can be very useful for filtering and/or ranking SLiM occurrences. Nevertheless, users should always be mindful that bioinformatics predictions almost invariably need additional validation to be sure of function, even where an estimated confidence in the predictions is returned. False positives can occur purely by chance or they might have a different function from that being sought. Variations of dibasic [KR][KR] motifs, for example, form the core of five different cleavage motifs in ELM as well as several targeting motifs [9]. These latter false positives are particularly hard to identify because they are probably under very similar structural and evolutionary constraints to the motif of interest.

There are essentially three strategies that can be applied to predicted SLiM occurrences. Firstly, contextual data can be simply provided to users, allowing them to weigh different lines of supporting evidence using specialist knowledge and human judgement. Secondly, features can be scored/weighted to produce a final metric for each occurrence, by which they can be ranked. Lastly, scores and/or context features can be used to reject certain occurrences outright. Where sequence/structure context is used, filters are often applied to the input data prior to the motif search, which is more efficient. These are clearly not mutually exclusive approaches and tools will often filter the weakest predictions before ranking and/or reporting context for the remainder. Filtering itself is not an all-or-nothing affair and should be set to an appropriate level for downstream analysis and the relative tolerance of false positives versus false negatives. Scansite [22], MnM [10] and SLiMSearch2 [68], for example, have different stringency settings depending on how strictly the user wants to filter occurrences.

2.1.1 Sequence Space Considerations

The number of motif predictions returned by any algorithm will rely heavily on the sequence space searched, with longer/more proteins likely to return more hits to the motif regex/profile by chance. On the other hand, real instances will be missed if the sequences containing them are missing or excluded from the search dataset. Indeed, the selection of protein sequences is just as important as the choice of the search algorithm for most applications. This also applies to analyses of individual proteins. SLiMs have been implicated in functional differences associated with splice variation [35–37] and so limiting analysis to canonical sequences could miss potential occurrences. Similarly, SNPs could create or destroy SLiM instances and should be considered where relevant.

When trying to identify and/or rate predicted novel instances of known motifs in a limited number of specific proteins of interest, it is usually safer to err on the side of caution when masking sequence data and/or filtering instances. Because such predictions are tempered by circumstantial data that is not inherently part of the SLiM definition, it is generally a good idea to maximize the sequence space. When searching larger datasets, it is more normal to apply more stringent filters and restrict the number of returned results to a manageable number. If this can be achieved by masking the input data prior to analysis then efficiency can also be improved, which is particularly useful if additional data (sequence alignments, etc.) are created or analyzed for each instance. When looking for enrichment of SLiMs within a dataset, protein sequence selection and sequence space masking are even more important as they will affect any statistical assessment of abundance.

2.1.2 Protein Structure

The majority of SLiMs are found in disordered regions of proteins, at least in the unbound form [1]. Therefore, it frequently makes sense to screen out globular regions prior to motif prediction [41, 63, 71, 77]; although some true positive instances are likely to be erroneously discarded, the hope is that a much higher proportion of false positives will be removed and thus the resulting predictions will be enriched for real SLiMs. Typically, a disorder prediction program (reviewed in [78, 79]) is used to identify and screen out predicted globular regions (e.g., [41, 63]). IUPred [80] is particularly popular for disorder prediction in SLiM discovery because it combines reasonable accuracy with being freely available for academic use. No disorder predictor is completely accurate, so it is generally recommended to err on the side of overprediction when masking based on disorder. Whereas the default

IUPred disorder cutoff is a score of ≥0.5, for example, cutoffs of ≥0.2 [41, 63] or ≥0.3 [81] are typically used due to the observation that 80–90 % of known ELM occurrences would be retained by such thresholds [1] [80]. We have previously found that the default IUPred disorder cutoff of ≥0.5 correctly classified approx. 95 % of ordered residues in the DisProt database [82] but only approx. 50 % of disordered residues (i.e., it is conservative), whereas a cutoff of ≥0.2 correctly classified approx. 95 % of disordered residues but only approx. 50 % of ordered residues (i.e., is very relaxed) (data not shown). It should be noted that this analysis was performed on a very limited dataset, although similar figures are given for IUPred defaults on CASP data [78]. No systematic analysis of the optimum disorder prediction for SLiM discovery has yet been executed although our own testing of IUPred has indicated that a conservative cutoff of 0.2 gives the best trade-off between specificity and sensitivity for predicting occurrences of known ELMs (data not shown).

An alternative strategy is to mask out domains identified by a domain database, such as Pfam [55]. Whilst this can be effective and was employed by Neduva et al. in their landmark SLiM discovery paper [71], it must be done with caution. Not all domains in Pfam are completely globular and a small but significant proportion are completely disordered [83–85], an observation that is supported by the suggestion that 40 % of the domain folds in the consensus domain dictionary (CDD) [86] are unstable, rather disordered and should not be considered traditional domains [87]. Thus, carefree masking of domains could result in removal of some genuine SLiM-containing regions. As the notion of disordered protein domains as biologically functional and important regions continues to gain widespread acceptance, it is possible that more such domains could end up in domain databases. Combining domain prediction with disorder prediction and/or cross-referencing to a database of disorder domain sequences (e.g., [83]) should help to avoid such errors. ELM, for example, uses a structural filter that combines solvent accessibility and secondary structure [57] to complement disorder predictions by GlobPlot [88] and IUPred [80] and domain predictions from SMART [54] and Pfam [55].

Structural information about PPI is scarce but since it offers direct evidence, it is a valuable resource for SLiM prediction. The 3did database [89] has 462 DMI of known 3D structure (as of January 2014) and rules derived from such data have the potential for predicting new instances and even entirely new classes of DMI [90]. Although biological relevance cannot be established from structure alone it is particularly useful to define the interaction interface with high confidence. An early example of this is iSPOT [91], which uses structural data to estimate the propensity of an input sequence to bind PDZ, SH3, or WW domains. It stores frequency tables of residue–residue contact pairs calculated from PDB structures of peptide-domain complexes and uses these to score the interaction with each fragment of a defined length in the input sequence. More ambitious is PepSite [92], which models preferred peptide-binding environments for protein surfaces to assess whether a given peptide could bind a given globular domain. This, in principle, could be used to help assess whether a novel instance of a SLiM is able to bind the appropriate domain.

2.1.3 SLiM Conservation

Evolutionary conservation is good for ruling out motif instances that are unlikely to have functional importance; however, there is a major complication to using evolutionary conservation as a discriminator for SLiM discovery. The plasticity of SLiMs and their propensity to occur in IDR means that they are frequently not conserved to the same evolutionary depth as globular domains [1, 12]. Even when the SLiM is conserved, the low conservation of surrounding disordered residues can generate alignment errors [93]. High variability in evolutionary dynamics between different SLiMs and IDR further limits the use of absolute measures of sequence conservation, which are heavily dependent on sequence quality and availability as well as the background evolutionary rate (i.e., functional constraint) of the parent protein. As a result, conservation metrics trained on discovering globular domains tend to overlook SLiMs; SLiM discovery requires its own conservation metrics.

MnM uses a conservation score based on BLAST pairwise alignment scores of HomoloGene clusters [94] and introduced the idea of adjusting conservation scores of predicted SLiM occurrences based on the overall conservation of the full-length proteins [95]. Dinkel and Sticht extended this idea, using weighted percentage identities that were normalized to the global percentage identity of the parent proteins [96]. These adjusted scores were then calculated across increasing numbers of homologues (sorted by relatedness) and the final distribution of scores used to rank predictions. Chica et al. took a different approach and normalized conservation scores by weighting conserved occurrences according to evolutionary distance and then normalizing to the overall tree weighting for each parent protein to allow comparisons between proteins and alignments [56].

These methods still suffer from different proteins (and protein regions) having different distributions of homologues available and/or different functional constraints across the full-length protein, independent of SLiM constraints. The solution, introduced by Davey et al. [27], is to measure the conservation of SLiMs *relative* to a surrounding window of (disordered) residues. This "Relative Local Conservation" (RLC) approach successfully adjusts for both homologue number/distance and alignment quality; if alignment of the SLiM-containing region is poor compared to the protein as a whole, this should not affect the score. Likewise, RLC effectively normalizes homologue numbers and evolutionary distances. Because individual alignment column scores are used for the RLC normalization, methods that weight conservation according to evolutionary distance can also be incorporated into the method [27]. Furthermore, it is possible to use the distribution of RLC scores to assign a statistical probability to observing a given cluster of high RLC scores at a motif instance, which can be used for ranking predicted occurrences [68] and even directly predicting de novo SLiMs [97].

Notwithstanding the fact that alignment errors will disrupt conservation patterns, the possibility remains that genuinely functional SLiM instances have evolved recently and therefore show little conservation. Likewise, apparent conservation of a given SLiM instance may be a chance occurrence or the consequence of evolutionary constraint on a similar/nearby sequence pattern wholly independently of the SLiM of interest; SLiMs often co-occur [13, 60] and flanking residues can also show correlated evolutionary patterns [98]. Despite these limitations, evolutionary conservation can be a powerful tool when harnessed correctly. ELM [9] incorporates a conservation filter based on the tree-weighting method of Chica et al. [56], whilst SLiMSearch2 [68] uses RLC [27] to help rank and filter results. SLiMProb [25] can mask input data using a number of conservation schemes including RLC.

2.1.4 Use of Other Contextual Information

Where there is sufficient annotation, additional contextual information can be used to rank or filter results. This is exemplified by MnM 3.0 [10], which employs a number of filters that compare predictions to the known target of the motif using a tightly controlled semantic syntax framework [14]. This allows biological data such as gene ontology (GO) and protein/genetic interactions to be combined with homology and structural data to screen out false positives. At the highest stringency, the authors report 39 % retention of validated instances with zero false positives returned. Whether this holds true for all SLiMs is yet to be seen but it demonstrates the importance of contextual information when ranking or scoring motif predictions.

PPI networks are an obvious source of contextual information to help support or reject SLiM predictions. For motifs with known binding partners/domains, these data can be used directly to assess occurrences. iELM [58] predicts new instances of known motifs by mapping instances together with known binding domains onto PPI networks, which can be supplied by the user. There are also tools for interactively exploring SLiMs in the context of PPI networks: SLiMScape [99] is a Cytoscape plug-in that can directly run SLiMSearch predictions of known motifs [68] or SLiMFinder de novo SLiM prediction [100]. For novel motifs, enrichment in such datasets can be indicative of function. This is explored in the next section.

Features of known SLiMs can also be used to build predictors of novel instances using machine learning algorithms. The AutoMotif Service (AMS) [101], for example, trains artificial neural network pattern classifiers for the automatic prediction of PTM sites. Annotated (positive) instances are taken from UniProt and Phospho.ELM, while negative training data is randomly chosen from fragments of sequences with no known PTMs. The disadvantage of machine learning approaches is their tendency to be a "black box", making human understanding and assessment of individual predictions quite difficult.

2.2 Assessing SLiM Occurrences at the Dataset Level

Assessing SLiM occurrences at the dataset level is largely performed for one of two reasons: exploring dataset function through known motifs, or exploring possible motif function through motif distributions. The latter is frequently used to add weight to the de novo SLiM predictions discussed in Subheading 3. In practical terms, the main difference is how many different protein datasets and SLiMs are considered: either a single dataset of interest is searched with one or more SLiMs, or a single SLiM is assessed in multiple datasets that subsequently need to be sorted and ranked (and controlled for multiple testing). The latter exercise is usually the remit of specific tools, such as SLiMSearch2 [68], which will identify GO categories and IntAct PPI partners [102] enriched for a specific SLiM from whole proteome occurrence data. For the purpose of this review, we focus on the general case of assessing a single SLiM in a single dataset. Where multiple SLiMs and/or datasets are used, an additional multiple testing correction will be required.

2.2.1 Overrepresentation Statistics

The most common analysis is to assess a motif for overrepresentation in a particular dataset. The most frequently used statistical approaches for such assessment are the cumulative binomial distribution and the cumulative hypergeometric distribution. In each case, we are interested in the probability of observing k or more successes (motif occurrences) given n trials (positions/sequences in which a motif could occur), each with a probability of success p. The main difference between the two approaches is that the binomial distribution calculates a probability based on n trials *with replacement*, which means that each motif occurrence is deemed to be independent and drawn from an infinite population. This is the norm for SLiM prediction tools. The hypergeometric distribution, in contrast, models n trials *without replacement* given a finite population N. This would be more appropriate for situations in which the total number of motif occurrences in a full dataset was known and enrichment was being assessed for a sub-sample of that dataset, e.g., a single PPI dataset or GO term in a whole proteome. Where N is much larger than n, the binomial is a good approximation and more efficient to calculate.

There are essentially two levels at which motif occurrence probabilities can be considered. At the sequence level, k is the total number of observed occurrences, n is the number of discrete positions at which a SLiM could occur and p is the probability that a motif occurs at any given position. At the dataset level, k is the number of different proteins containing the SLiM, n is the total number of proteins and p is the probability of the SLiM occurring in each protein. In each case, k is normally quite obvious but n and p can vary in the way that they are calculated.

The biggest challenge is correctly modelling the background frequency distribution from which to derive the probability of occurrence per site/protein, p. Due to differences in amino acid composition, one has to consider whether to base expectations on protein-specific or dataset-specific amino acid frequencies. If residues have been masked based on evolutionary or structural information as previously described, there can be big differences between the masked and unmasked data. Alternatively, frequencies might be derived from a different "background" dataset, such as a complete proteome. At one extreme, if sequence biases are not correctly handled then returned motifs will simply represent this bias. SLiMs, for example, tend to occur in disordered regions [1] and so it is common to mask out globular domains and/or predicted ordered regions [41, 63, 71]. Disordered sequences are known to have a different amino acid composition to globular domains [1, 80] and thus if full-length protein sequences are used for the background expectation, there will be a tendency to overpredict motifs because all disordered amino acids are enriched. At the other extreme, if one masks the sequences perfectly so that *only* motif sequences are unmasked, the observed amino acid frequencies will be biased *because of* the motif occurrences, which might make the motifs themselves appear uninteresting. This is especially true for low complexity motifs, which predominantly feature the same amino acid(s) in multiple positions. In general, the sequence space is considerably larger than the motif instances and so the assumption is that the motif does not bias the background.

Once the probability of a given motif occurring at any given position can be calculated from a background frequency, the number of such positions must be taken into consideration. For sequence data, the probability of motif occurrences is clearly related to the size of the sequence space being searched, which in turn determines n. It is here that conservation and disorder masking make a big difference by reducing the number of sites at which a motif can occur by chance. To a first approximation, n is equivalent to the number of unmasked amino acids in the protein.

For datasets of multiple proteins, the probability of occurrence in each protein can be calculated as just described. Alternatively, amino acid frequencies can be bypassed by empirically estimating per-protein probabilities based on occurrences in a background dataset. This approach must be used with caution; there may be biases in protein composition in addition to any amino acid composition bias. The main problem is the presence of homology, which makes motifs more likely to have extreme distributions than would be expected if all proteins were unrelated, making *p* hard to estimate. Problems with homology also apply to the dataset of interest, for which the statistical model assumes that the *n* proteins are independent. Evolutionary relationships between proteins can heavily skew statistics by breaking this assumption of independence, regardless of how *p* is calculated. To counter this, SLiMProb uses the SLiMChance probability model of SLiMFinder [41], which in turn uses the "Unrelated Protein Clusters" (UPC) correction for evolutionary relationships introduced by SLiMDisc [77]. Under this model, BLAST [103] is used to identify homologous proteins, which are then clustered such that no protein in an UPC has detectable homology with a protein in another UPC. Dataset size (*n*), motif support (*k*) and the probability of SLiM occurrence (*p*) are then calculated using the UPC rather than individual proteins. The importance of this correction cannot be overstated: statistical models that ignore sequence homology cannot be trusted unless the input has similarly been purged of homology. SLiMProb also calculates enrichment for proteins without evolutionary filtering (i.e., assuming evolutionary independence) and for the overall number of occurrences across all sequences (i.e., *k* is all occurrences, *n* is the entire sequence space and *p* is the probability per site), which enables the effect of the correction to be examined.

2.2.2 Underrepresentation Statistics

Although it is less common, it can also be interesting to assess whether a SLiM is *underrepresented* in a given dataset. The statistics for this are essentially the same, except that the main concern is the probability of seeing *k or fewer* occurrences given *n* and *p*. It has been observed that false positive occurrences of some motifs are underrepresented in certain datasets [104]. Combining overrepresentation and underrepresentation statistics could therefore prove an interesting way to explore the evolutionary and, by proxy, functional dynamics of a SLiM within a proteome. In particular, one would expect an overrepresentation of conserved instances where the SLiM is functionally important but an underrepresentation of non-conserved instances where they might disrupt signalling and are therefore subject to negative selection. This has clear implications for using overrepresentation in de novo SLiM prediction and is discussed in more detail below.

3 Computational De Novo SLiM Prediction

Many of the considerations for predicting instances of known motifs also apply to the task of predicting SLiMs de novo. Understanding critical features of known SLiMs has allowed the establishment of a set of rules helpful to find new motifs. The task is clearly more complex when the nature of the motif is not known. It is not simply a question of where functional instances of the SLiM might occur; selecting the right tool for the job depends on the data available, the nature of the motif (length, conservation, induced structure (if any), whether specific amino acid side chains/PTM will be involved, etc.) and the level of confidence that a SLiM is present in the data at all. Where the latter is unknown, estimation of the significance of results is essential. This section will explore these issues and how to make the best use of the available data. Recommendations will be made where possible (*see* also Table 5) but it should be noted that de novo SLiM discovery is still a developing field and there may not (yet) be an obvious "best" approach in all situations.

It is important to remember that de novo SLiM discovery will rarely return the SLiM precisely as it would be defined by in-depth study. This is because the number of instances in nature is frequently insufficient to have fully explored sequence space through evolutionary time. With the exception of very specific motifs, such as the integrin-binding RGD motif, it would be impossible to return the complete motif definition given the sequence data available (Table 6). Even then, it is possible that additional subtle features of the SLiM are yet to be discovered. Because SLiM discovery tools are mining the strongest signals, they will generally return a simpler version of the motifs and/or include some extraneous flanking residues. One should remember this when analyzing the results of any SLiM prediction: the real SLiM (if there is one!) is likely to be a somewhat refined version of the pattern returned by the program. It is also important to remember that not all occurrences in the input data are necessarily going to be functional. Depending on the planned follow up, it might be necessary to rank and/or filter those occurrences using the same techniques as previously discussed for predicting known motifs. Likewise, the data used for SLiM prediction might not include all of the functional instances; it can be useful to perform a large-scale analysis of the distribution of the predicted SLiM, both to identify additional instances and provide insight into whether the motif is genuinely associated with the dataset from which it was predicted.

Tools for de novo motif prediction from sequence data can be broadly classified depending on their goal:

1. Alignment-based algorithms aim to best describe a single motif based on an alignment of motif occurrences.
2. Alignment-free methods aim to interrogate multiple sequences to identify a new common feature.

Alignment-based methods clearly need to use additional information when compared to alignment-free methods in order to

Table 6
SLiMMaker consensus motifs from annotated ELM instances for top 20 ELMs ranked by instances in ELM

ELM	ELM regex definition	Refined SLiMMaker regex[a]	N[b]
LIG_WRPW_1	[WFY]RP[WFY].{0,7}$	[WY]RP[WY]	93/95
LIG_EH_1	.NPF.	NPF	88/88
LIG_AP2alpha_2	DP[FW]	DP[FW]	54/54
LIG_PDZ_Class_1	…[ST].[ACVILF]$	[ST].[LV]$	41/48
MOD_NMyristoyl	^M{0,1}(G)[^EDRKHPFYW]..[STAGCN][^P]	^MG[AGNQS]..[AGS]	38/48
MOD_SUMO	[VILMAFP](K).E	[FILV]K.E	43/45
CLV_C14_Caspase3-7	[DSTE][^P][^DEWHFYC]D[GSAN]	[DST].[LPTV]D[AGS]	25/39
LIG_SUMO_SBM_1	[ILV](.[ILV]|[ILV]|[ILV].)[ILV][STDE]{1,10}	[ILV][ILV][DIL][DLS][DST]	27/39
LIG_CtBP_PxDLS_1	(P[LVIPME][DENS][LM][VASTRG])|(G[LVIPME][DENS][LM][VASTRG]((K)|(.[KR])))	P[ILM][DN]L[RS]	19/32
LIG_Rb_LxCxE_1	[LI].C.[DE]	L.C.[DE]	31/32
LIG_WW_1	PP.Y	PP[AEP]Y	21/28
MOD_PKA_2	.R.([ST])[^P]..	R.S	27/28
TRG_PEX_1	W…[FY]	W..[DEQ][FY]	23/27
TRG_NLS_MonoExtN_4	(([PKR].{0,1}[^DE])|([PKR]))((K[RK])|(RK))(([^DE][KR])|([KR][^DE]))[^DE]	[KPR].[KR].[KR]	18/26
LIG_PTAP_UEV_1	.P[TS]AP.	P[ST]AP[LPQS]	20/25
MOD_PKA_1	[RK][RK].([ST])[^P]..	[KR]R.[ST]	23/25
DEG_SCF_TIR1_1	.[VLIA][VLI]GWPP[VLI]…R.	QIVGWPPVRSYRK	3/24
LIG_NRBOX	[^P]L[^P][^P]LL[^P]	L..LL	24/24
MOD_CMANNOS	(W)..W	W[GS][EPS]W	12/24
MOD_LATS_1	H.[KR]..([ST])[^P]	H.R..[ST]	21/23

[a] Product of iterative SLiMMaker regex construction from annotated ELM instances with default settings: each variant in an ambiguous position must be present in at least 3 sequences; max 5 variants per ambiguous position; during iterations, 75 % sequences must match position to be non-wildcard

[b] The number of annotated ELM instances matching the refined SLiMMaker regex

constrain the motif search. Such methods are more restricted in terms of potential applications but can use approaches that are unsuitable for less constrained alignment-free data.

3.1 Alignment-Based (Divergent Evolution) Methods

Building on the success of protein domain prediction/definition, some de novo SLiM methods attempt to identify SLiMs on the basis of signals of evolutionary conservation among homologous protein regions, i.e., purifying selection acting at functionally important sites during divergent evolution. The challenge is that globular domains dominate this signal and so SLiM-specific models of evolution must be applied. Recently, methods have harnessed the power of the "Relative Local Conservation" (RLC) method discussed in Subheading 2 [27]. SLiMPrints uses a statistical model of clustered RLC-conserved residues to identify evolutionary signatures of SLiM occurrences and return them as regex patterns of fixed and wildcard positions [97]. A similar approach has also been taken using phylogenetic hidden Markov models (phylo-HMM) to search for locally conserved sequences in unstructured regions [105]. The nature of profile-based methods make them particularly suitable to capture the evolutionary constraints of homologous sequences; another example is MFSPSSMpred [106], which incorporates local conservation scores from multiple sequence alignments into a support-vector machine model of MoRF sequence features to predict novel SLiMs/MoRFs. These methods have the advantage of being able to identify singletons (i.e., motifs with a single known instance) but additional data and/or experiments will be required to predict the function of any SLiMs that are discovered.

Alignments can be based on function rather than homology. Motif-x, for example, is an alignment-based method that is actually modelling convergent evolution by aligning otherwise unrelated sequences around key residues such as phosphorylation sites recognized from mass spectrometry data [107]. Fixed position motifs are constructed from a window either side of the aligned residue. SLiMMaker will similarly generate a consensus regex SLiM (with ambiguity) from a set of aligned peptide sequences, whether they are homologous or not (Table 6).

3.2 Alignment-Free (Convergent Evolution) Methods

One of the most common and successful approaches for de novo SLiM prediction is the interrogation of multiple different proteins for shared sequence patterns. Unlike most alignment-based approaches above, these methods are modelling *convergent* evolution, i.e., the independent origin of shared motifs on unrelated protein backgrounds. There are a number of potential sources for such protein sequences [11] but the most common are PPI data [108] and functional classifications such as GO. In each case, methods are generally seeking either (a) the most abundant patterns in the data, or (b) the most enriched patterns versus a background expectation.

The latter are generally more effective due to inherent biases in amino acid frequencies but they are reliant on good background models to calculate the expected motif abundance by chance.

One of the earliest dedicated de novo SLiM discovery tools was Pratt [109, 110], which updated and extended an earlier approach by Neuwald and Green [111]. Although Pratt is an alignment-free method, it was originally designed with divergent sequence motifs in mind, such as PROSITE family descriptors [18], which is reflected in the parameters. Pratt is still useful for returning a ranked list of motifs that include amino acid and wildcard-length degeneracy. It is designed to find patterns that occur in the majority of sequences and does so efficiently. The algorithm can be very slow when searching for patterns present in only a few sequences, particularly when the motifs are small. Output is highly dependent on parameter settings and it does not return a statistical significance for predicted SLiMs. Furthermore, because it was designed with sets of homologous proteins in mind, there is no evolutionary filter to model convergent evolution. As a consequence, although still available at EBI, Pratt is not recommended for general de novo SLiM discovery. A better alternative is SLiMFinder [41] (below), which returns Pratt-like flexible patterns but is optimized for convergent evolution and also provides an estimate of statistical significance for predictions.

Another notable early tool is TEIRESIAS [112], a general text pattern finding tool that has been widely applied to the problem of de novo SLiM discovery and served as the inspiration or basis for several tools that followed. TEIRESIAS can return "degenerate" motifs with site-specific variability through equivalence sets of residues, i.e., sets of amino acids that can co-occur in ambiguous positions. Any user-defined equivalence sets may be used but it is most common to group amino acids that share physicochemical properties. Given a set of protein sequences, a scanning phase takes place to collect all putative motifs of given length, proportion of defined sites and support in the dataset. These are then combined recursively into longer patterns with enough support, keeping the efficiency of the algorithm by discarding patterns that are less specific versions of others, while accounting for ambiguity by treating all residues in the same set as equals. Homologous sequences in the input dataset can inflate support for certain motifs, which can also result in very long run times and massively increase the number of patterns returned. Nevertheless, it is still sometimes used for baseline performance comparisons in methods benchmarking.

3.2.1 Methods Correcting for Evolutionary Relationships

One weakness of the early methods is their failure to consider evolutionary relationships in the data. This is important, otherwise large regions of homology will be returned as motifs in a potentially misleading fashion. The first method to correct for this was DILIMOT [71, 113], which filtered homologous regions and kept a single representative for analysis. Whilst this kept the underlying TEIRESIAS pattern discovery step efficient, it has the slight disadvantage of removing sequence variants that might better reflect the core SLiM. Another concern is that weakly homologous regions flanking those removed by the filter might remain in the data and bias results. Nevertheless, DILIMOT was a major advance in de novo SLiM discovery and its application to human, fly, nematode and yeast PPI data was a landmark paper in the field.

SLiMDisc (Short Linear Motif Discovery) [42, 77] was developed around the same time as DILIMOT but took a different approach to correcting for evolutionary relationships. Instead of filtering homologous sequences, motifs were given a heuristic score that was based on their homology-corrected support and information content (i.e., length and degeneracy). Three different correction methods were tested. The best performance was achieved by scaling motif support using a "Minimum Spanning Tree" (MST), which would produce a corrected support from 1 to N, where N is the number of proteins in which the motif is found. If all N proteins were identical, MST would scale support to equal 1. If all N were unrelated, support would be N. Like DILIMOT, SLiMDisc used TEIRESIAS for underlying pattern discovery and was essentially an add-on for filtering and ranking TEIRESIAS output. The original SLiMDisc scoring was subsequently modified in the webserver implementation using "SLiM Pickings," which weighted the original SLiMDisc score according to the ratio of observed versus expected support, corrected for evolutionary relationships and amino acid frequencies of the input data. Later releases of the webserver have seen TEIRESIAS pattern finding replaced by the SLiMBuild algorithm of SLiMFinder [41, 100].

There are two main drawbacks of the SLiMDisc/DILIMOT approach. Firstly, scores are not directly comparable between datasets. Secondly, whilst the methods are very good at returning real motifs among the top-ranked patterns in the output, there is no way of assessing how likely it is that a given data had *any* genuinely overrepresented motifs. SLiMFinder (Short Linear Motif Finder) [41, 100] overcomes these two problems by carefully controlling the motif space during motif construction using its own SLiMBuild algorithm in place of TEIRESIAS. This motif space is then used by the SLiMChance algorithm to robustly, if somewhat stringently, estimate the significance of overrepresented motifs. SLiMChance uses the binomial distribution as described for SLiMProb in Subheading 2 with an additional multiple testing correction for motif space. This again uses the cumulative binomial distribution, where k is 1 (a single successful motif), n is the total number of motifs in the motif space, and p is the individual motif's overrepresentation probability. These solutions enabled large-scale analysis of tens of thousands of human protein datasets [63]. SLiMFinder also introduced the capability to return motifs with flexible-length runs of wildcard positions, which are important for some motifs.

It should be noted that correcting for evolutionary relationships is not always possible. Sometimes, numerous overrepresented motifs will be returned from the same (sub)set of input proteins even if there is no BLAST-detectable homology; BLAST can sometimes miss homology in short and/or low complexity proteins. For this reason, it is always advisable to manually visualize the context of significant motifs. This is easier with tools like SLiMFinder that output such alignments for visualization as part of the results. In extreme cases, sequences may have diverged to the point that conserved SLiMs are the only detectable homology. This will be impossible to distinguish from convergent evolution but should not affect the performance of SLiM discovery tools. There can also be problems with very large datasets. The "Unrelated Protein Cluster" (UPC) method employed by SLiMFinder works by clustering proteins via BLAST homology connections such that no proteins in one UPC will have detectable homology with any proteins in a different UPC. This does not necessarily mean that all the proteins in a cluster will share sequence homology: if protein A is homologous to B and B is homologous to C in a different region/domain, A and C will be grouped in the same UPC despite having no direct homology. For large datasets of multi-domain proteins, such as mammalian proteomes, this can result in a substantial proportion of the data (in the order of half the proteome) clumping together into a single giant UPC (data not shown).

3.2.2 Profile-Based Methods

Another popular set of programs for motif discovery is the MEME Suite of motif-based sequence analysis tools [72]. MEME [114] was developed originally as a method for novel motif discovery in DNA sequences. Genomic sequences are still its main focus but it can be used for finding signals in any biological sequence and has been applied to SLiM discovery. MEME uses ungapped PSSMs to represent motifs as extracted from an unaligned set of input sequences. It assumes that each sequence in the starting dataset contains an instance of the motif and employs the expectation-maximization algorithm [115, 116] to find motif patterns with high likelihood. Haslam and Shields [81] have found that MEME cannot perform as well as SLiMFinder, which is based on regular expressions, unless evolutionary weighting and local conservation filtering are applied. In that case both approaches are shown to be complementary, with some motifs being only returned by one of them, suggesting that their joint application can lead to an extended coverage of the results. It should be noted, however, that MEME does not return a significance estimate akin to SLiMChance and therefore this analysis was based on the ranks of positive predictions. Other tools in the MEME Suite extend the reach of the main algorithm by allowing searches, comparisons, and functional predictions for DNA and/or protein motifs. Notably, motif discovery is improved with GLAM2 [74] by incorporating gaps of flexible length in the definition of motifs. Given a set of sequences believed to share one or more motifs it will perform a gapped alignment of them to find, score and rank all conserved patterns. Like MEME, GLAM2 is optimized for DNA motif discovery.

NestedMICA [117, 118] uses different probabilistic models to represent motif-carrying and non-motif fragments in the input sequences, identified through a nested Sampling strategy. The motifs are represented by a set of profiles extracted from the provided data and a predefined, non-homogeneous model of "uninteresting" background information serves as reference. These are all combined in an HMM model that is updated in each iterative step of nested sampling after discarding a certain fraction of sequence space and until the likelihood of the resulting motif profile is maximized. The output of NestedMICA is a profile for each motif, displayed as a sequence logo. An assessment of its performance against MEME over a purposely built benchmark dataset showed that NestedMICA can retrieve more true positive hits while reducing the false negatives at the same time [118]. However, although the information content values in each column of the logo give a hint on variability and conservation, the program does not provide any significance measure of motif support.

3.2.3 (l, d) Motif Searches

One common motif formulation for general de novo motif discovery is the "(*l*, *d*) motif search" (LDMS) (also known as a "planted motif search" or "(*l*, *d*) challenge problem"). LDMS algorithms search for all motifs of total length *l* (including wildcards) with up to *d* mismatches (i.e., 0-*d* wildcards). There are many LDMS algorithms and programs; recent examples include qPMS7 [119, 120]. LDMS algorithms are generally developed and tested for DNA motif discovery, but are rarely benchmarked or optimized for protein searches, which have very different constraints and criteria. Developments frequently concentrate on computational performance (i.e., speed) but often overlook important biological considerations, such as evolutionary relationships that can bias results, low support in the input data, or the possibility that there may be *no* real motifs in the data to find, which results in a lack of statistical significance. For these reasons, LDMS tools are not generally recommended for de novo SLiM discovery.

This is not to say that no LDMS algorithms are useful, but it is inadvisable to apply an algorithm to protein motif prediction if it has only been benchmarked on DNA data. Without this estimation of statistical significance, applying an LDMS algorithm to a dataset that may not contain a motif (of the "(*l*, *d*)" nature being sought) is almost guaranteed to generate false positive predictions. In principle, LDMS motif space could be modelled for a statistical assessment of motif support, although it is often over-constrained by the need to fix the *l* and *d* parameters prior to searching. Another feature of LMDS algorithms that can be problematic is that they do not generally return a motif in the way defined in Subheading 1. Instead, the output is a consensus sequence and a list of variants with mismatches. Because these mismatches can occur in different positions in each motif instance, it can be difficult to generate a regular expression that captures the variability, although they could conceivably be coupled to an alignment-based algorithm to construct a sequence profile. If the natural incorporation of mismatches from the consensus "motif" could be correctly incorporated into a robust statistical framework like SLiMChance [17, 41], LDMS algorithms could yet prove useful for de novo SLiM discovery in real biological data.

3.2.4 Co-occurrence Methodologies

Correlated Motif Mining (CMM) methods suppose that motifs, being short and flexible, may be directly involved in interactions between larger domains. D-MOTIF and D-STAR [121] are the exact and the approximate versions of an algorithm built on the LDMS model to find instances of two motifs that are correlated in the same interaction in a PPI network. Leung et al. raised adequacy and scalability issues in D-MOTIF and D-STAR and proposed an alternative model to find motif pairs based on fast clustering heuristics that they implemented as MotifCluster [122]. Another CMM method, SLIDER [123], incorporates structural data and maps correlated LDMS occurrences onto PPI interfaces. In addition to the underlying LDMS drawbacks, the main issue with CMM is that there are no clear examples of SLiM-SLiM PPI and it is highly likely that either or both motifs returned are actually structural/family signature motifs of domain-based interactions. Another weakness of this method is that it cannot identify motifs that all interact with the same single partner. In addition, whole interactome data is required for the prediction, which limits application.

FIRE-pro [124] is another CMM method that uses mutual information (MI) to discover motifs whose presence/absence correlates with a biological feature of the proteins in question. FIRE-pro first builds gapped *k*-mers (fixed position motifs with *k* defined positions separated by runs of 0–3 wildcard "gap" positions) and calculates their mutual information with the biological classification of the proteins in the dataset. Motifs that tend to be present in proteins that are positive for the biological feature of interest (e.g., GO category or PPI partner) and absent in negative proteins have a high MI score. This is then compared to randomized feature classification and only those motifs with higher MI than 10,000 randomizations are retained. More informative descriptions of the significant motifs are ultimately informed by subjecting those to a greedy search of variants that increases their degeneracy. FIRE-pro does not use protein disorder information and might therefore return structural motifs; a possible future improvement would be to couple FIRE-pro with disorder and RLC masking. Although FIRE-pro does filter evolutionary relationships, the BLAST

E-value used is extremely stringent (1e-50) and therefore it is highly likely that homology will be influencing some of the MI associations. One also needs to be very careful of complex PPI relationships and PPI-GO correlations that could give rise to false associations, particularly in multi-domain proteins. With those caveats in mind, the high efficiency of FIRE-pro makes it suitable to analyzing proteome-scale data to discover, rediscover, and make an initial functional prediction of SLiMs.

3.3 Sequence Property/Feature Methods

Not all de novo SLiM discovery tools make predictions based on sequence specificity. Whilst not the focus of this review, it is useful to briefly highlight a few of these other methods. Frequently, these approaches will complement a sequence-based approach and can provide useful corroborating evidence regarding the nature/importance of a given site. Alternatively, they might be useful to preprocess data for sequence-based predictions and or rank/filter results as explored in Subheading 2. ANCHOR [125, 126], α-MoRF-PredII [44, 127], MoRFPred [128], and MFSPSSMpred [106] use signs of propensity for structure within an IDR to predict potential SLiM- or MoRF-containing regions. SLiMPred [129] takes a more flexible machine learning approach and uses annotated ELM instances and structural, biophysical, and biochemical attributes predicted from the primary sequence to build a bidirectional recurrent neural network that generates a per-residue probability of being part of a SLiM. Output can then be scanned for clusters of SLiM-like residues. Although these methods do not generate motifs as such, they have the advantage of being able to identify interaction sites that lack the sequence specificity of SLiMs and/or where additional data (e.g., homologues, structures, or PPI) are unavailable. Where homologues are available, alignment-based tools can then be used to generate a motif consensus or profile for the identified region.

Efforts have also been made to predict novel SLiMs directly from structural data in the Protein Data Bank (PDB) repository [130]. D-MIST [131] is a profile-based method that uses structure-derived binding profiles to interrogate sequence databases for novel PPI. It first extracts motifs known to bind the same domain from structural complexes where the latter is present. The motifs are then used to seed a Gibbs sampling search of similar sequences from empirical binary interactions, from which PSSMs can be constructed and used to find other proteins with a similar interface. Similarly, SLiMDIet [132] has been developed to identify SLiMs in the binding interface of solved structures of PPI complexes in PDB. Pfam domain binding interfaces are clustered by structural similarity and the residues belonging to the domain face and the partner face in each cluster are then aligned. Based on the contacts of the interaction, SLiMs are extracted as flexible, gapped PSSMs, and their statistical significance assessed through PPI data.

Stein and Aloy [90] took a more focused approach and specifically modelled the features of known SLiMs from solved DMI structures in PDB [89], identifying a signature stretched and elongated structure that was characteristic of DMI peptides. Using machine learning and contextual filters, they then predicted novel DMI from PDB PPI, deriving consensus patterns using SLiMFinder [41] where possible. As with SLiMDIet, significance of predicted motifs was assessed using overrepresentation in PPI data. These methods show a lot of promise and are likely to become increasingly useful as the number of solved DMI continues to increase.

3.4 Statistics for De Novo SLiM Discovery

When it comes to motif prediction (and benchmarking of SLiM discovery algorithms) an indication of significance through testing on data without a genuine signal is crucial. This is because one of the primary challenges for de novo SLiM discovery is determining whether there is a motif to be found at all. There are good discussions of motif statistics for both regex [17] and profile [19] approaches elsewhere. Instead, this review concentrates on some of the biological and practical considerations that are likely to be pertinent, whatever the specific statistical model employed.

3.4.1 Sequence Space Considerations

The considerations for sequence space when searching for de novo motifs are much the same as previously discussed for making dataset-level overrepresentation assessments for known SLiMs. The main difference is that the additional multiple testing corrections for motif space in de novo searches dramatically reduce significance levels and thus any loss of signal by erroneously removing real instances (either by masking them out or excluding the parent sequence from the dataset) can have much stronger consequences than for the prediction of known motifs. As a result, whereas known motif prediction has a tendency to err towards stringent filtering, de novo prediction generally needs to maximize the available signal, even if that comes at the cost of increased noise. For a discussion of some approaches in dataset construction that can influence the signal–noise ratio, *see* [11].

Conserved Versus Non-conserved Motif Occurrences

Motif enrichment is potentially confounded by two opposing trends occurring in the dataset: enrichment for functional motifs [63, 71] and depletion of non-functional motifs [104]. SLiM-mediated PPI are frequently cooperative and/or competitive [8, 133] and therefore having competing sites of interaction in the wrong place at the wrong time could upset the delicate balance of signalling. Random occurrences of a SLiM in proteins that (could) interact with a given SLiM-binding domain are probably under negative selection [104]. These conflicting signals could obviously hamper SLiM prediction as the true random background would be smaller than that modelled. In reality, the number of motifs expected to occur by chance is usually quite small and therefore the loss of these from the actual signal is hopefully not too detrimental. The fact that SLiM prediction by overrepresentation works for many known examples [41, 63] supports this notion. Nonetheless, the observation that current SLiM prediction methods seem to be on the cusp of successful SLiM discovery in many cases implies that slight increases in signal or decreases in noise could be the difference between a SLiM being significantly overrepresented or not. Correct modelling of negative selection could potentially push some of these datasets into the detectable signal–noise range.

3.4.2 Motif Space Considerations

The main difference between searching for known SLiMs and de novo discovery is the large multiple-testing correction that needs to be done when the SLiM is unknown. In essence, any possible motif that could have been constructed by the de novo method could be overrepresented in the data. One of the major advances of SLiMFinder [41] was the SLiMBuild algorithm that tightly controlled the building of the motif space and thus enabled an exact calculation of the number of possible motifs. TEIRESIAS [112], which underpinned earlier algorithms such as DILIMOT [71, 113] and SLiMDisc [42, 77], does not control the motif space in the same way, which makes it hard to estimate how many different motifs are actually being tested. There is clearly a trade-off in the selection of SLiMBuild parameters, such as the maximum number of wildcards between defined positions (set to 2 by default): increasing the number of motifs increases the chance of the correct motif being within that motif space but also increases the size of the significance correction that is required. It should be noted that there might be ways of improving performance by altering the motif-building rules. Many motifs predicted by Neduva et al. [71, 113], for example, have three consecutive wildcards and would thus be missed by SLiMFinder defaults. It could be that by restricting the total number of wildcards akin to LDMS algorithms, rather than constraining the wildcards between pairs of defined positions, a more appropriate motif space could be constructed. Other approaches could reduce the motif space to focus on specific motif types. SLiMFinder, for example, includes an "alpha helix" mode that considers the helix periodicity and only searches for motifs in positions i, $i+3/4$, $i+7$, although this is yet to be benchmarked.

Motif Independence and Clouding

The statistics for SLiMs with different numbers of defined positions are generally kept separate as clearly they are not independent. PxxP, for example, is a sub-motif of PxxPx[KR] and their frequencies will clearly be related. Unfortunately, when such overlapping motifs are returned, it is currently impossible to tell whether the shorter is enriched because of enrichment of the larger or vice versa. Early attempts with SLiMFinder to incorporate the frequency of shorter versions of the returned pattern to assess significance were not very successful (data not shown) but it is a potential improvement to the statistical model that could be considered in future. Instead, SLiMFinder groups overlapping motifs (based on occurrences in proteins, not pattern definitions) into "clouds," allowing the user to rapidly identify different variants of the same general motif prediction [41]. Similarities between different SLiMs can also be identified a posteriori using CompariMotif [69]. Aligned occurrences of SLiMFinder clouds could be subsequently passed through an alignment-based tool, such as SLiMMaker or MEME, to give a more complete definition of the cloud.

A particular challenge to the statistical assumption of motif independence is the treatment of ambiguity. Ambiguous motifs are clearly not independent from the variants used to build them, thus increasing the motif space by all possible ambiguous motifs would unfairly and dramatically inflate the multiple testing correction. The current implementation of SLiMChance therefore ignores ambiguity when calculating and correcting for the size of the SLiMBuild motif space. For a complete motif space and limited equivalencies, this does not seem to affect the model too badly. The affects have not been well modelled, however. If too many equivalence sets are used it could result in inflation of significance for ambiguous motifs. In general, while ambiguous motifs are often more informative than pure fixed position motifs, confidence that they represent reliable predictions can be substantially increased if there is also a pure fixed position motif returned in the same cloud.

Altered Alphabets and Specified Amino Acids

One way to reduce the size of the motif space being searched is to reduce the alphabet. This can be achieved by combining certain amino acids with similar properties. The utility of this approach is not clear, however. There is an obvious trade-off between reducing the motif space and increasing the probability of given patterns occurring by chance by increasing the corresponding frequency of the new characters. This is seen in the difference between protein and DNA motifs: the latter need to be longer and/or more abundant to achieve significance. A second problem with this idea is that the biological justification is not clear. Whilst one could imagine combining lysine and arginine as positively charged amino acids, for example, they are not equally used by known motifs [1].

A more powerful way (in principle) to reduce motif space is to mask out certain amino acids that are unlikely to be of interest. Alanine and glycine, for example, are generally considered to be quite boring. Because this reduces the sequence search space as well as the motif space, it gets around the problem of motifs becoming more likely to occur by chance. That said, it should be noted that all 20 amino acids are found in the defined positions of at least one known motif, so such filtering should be applied with caution. SLiMFinder includes an option to mask out specific amino acids but this has not been benchmarked.

An alternative that is less extreme, and often easier to justify biologically, is to focus on motifs that contain a specific amino acid, such as a tyrosine if tyrosine phosphorylation is known to be important. This does not reduce the motif space as dramatically but can ease interpretation. An obvious application for such reductions would be the prediction of PTM sites. Indeed, in this instance it is possible to *expand* the alphabet by encoding modified residues with a twenty-first letter (e.g., Z) and then (optionally) specifying that motifs should have this letter. Although this is an option in SLiMFinder, we are not aware of any published work exploring or using this feature. A possible exception is the successful discovery of terminal motifs by SLiMFinder, which adds N-terminus (^) and C-terminus ($) characters to each protein before expanding the protein alphabet to include them [41, 63].

Controlling Motif Space with Defined Queries

The other way to reduce the motif space is to build it on a specific sequence (or set of sequences) rather than looking at all possible motifs. This is the basis of QSLiMFinder ("Query SLiMFinder" [140], Table 1), in which the motif space is built on a specific "query" protein sequence. Query motifs are then assessed for enrichment in another set of proteins that, for example, share a common PPI partner using the basic approach of SLiMFinder. Clearly the query sequence(s) used for building the motif space cannot be included in the search space itself as this would artificially inflate the support for those motifs in the data and thus there is a trade-off between reducing the motif space multiple testing and loss of signal in the data. QSLiMFinder can substantially improve search sensitivity over SLiMFinder where the query protein/region is quite small, e.g., a short binding region has been identified (data not shown). One caveat is that ambiguity cannot be usefully included in the statistical model: in order to be valuable, motif variants outside of the query must be included but these inflate the motif search space by an unknown amount. One solution is to return ambiguous motifs but only pay heed to those that also have a significant fixed-position pattern returned in the same cloud.

3.5 Low Complexity Motifs

Low complexity motifs are motifs that are dominated by a small number of amino acids. Examples include proline-rich motifs, serine-rich motifs, RG and RS repeats, and patches of positive or negative charge. Low complexity regions have a tendency to return a lot of similar motifs, especially if variable-length wildcards are used. Masking low complexity sequences can avoid this but at the risk of missing genuine low complexity motifs. As some low-complexity motifs are certainly functional, this trade-off is largely going to be determined by the scale of the analysis being performed. Large-scale analyses will probably want to mask low complexity regions more stringently, as the probability of throwing together some proteins that share low complexity regions by chance will be high. Focused small-scale studies, on the other hand, should be more cautious. When such motifs are returned, the question must be asked: does the low complexity motif simply reflect a sequence bias, or does any such sequence bias reflect a high frequency of functional low complexity motifs in the dataset? As with most bioinformatics predictions, SLiM discovery tools cannot themselves answer this and additional evidence must be considered.

Large-scale application of DILIMOT to PPI data showed a marked tendency to recover proline- and serine-rich motifs [63, 71]. This could reflect a genuine bias towards these residues in SLiMs or might be a reflection of their occurrence in low complexity proline- and serine-rich regions of proteins, which are conserved (at the level of amino acid enrichment) between species. A large-scale analysis of human interactome data using SLiMFinder did not find the same degree of bias [63]. As this analysis masked out very low complexity regions and used motif conservation (as opposed to rediscovery), the implication is that the enrichment in the Neduva et al. study is largely due to low complexity regions. Of course, such low complexity regions are presumably functional and are genuinely enriched in the data: the question is whether they are enriched because they mediate the PPI of interest, or whether they are a different recurring feature of proteins that some methods are particularly sensitive to finding.

3.6 Predicting SLiMs from Short Peptide Data

The importance of correcting for evolutionary relationships has been stressed in the preceding sections. Sometimes, such correction is neither appropriate nor necessary because the input data does not have evolutionary relationships to worry about. Examples of this are phage display and peptide libraries, which sample a large sequence space and select for short peptide regions that can bind a desired partner. These techniques can be very useful for SLiM discovery as they can substantially increase the number of motif occurrences. Indeed, they can potentially be used to identify motifs in singleton interactions where enrichment in true PPI is not possible. The proportion of input sequences assumed to contain the motif is also high, which makes profile methods such as those in the MEME Suite [72] popular for such applications. SLiMFinder can be used to predict significantly enriched regex motifs from peptide data (*see* example 3 in the original paper [41]) but the background amino acid frequencies will need to be corrected to represent the pre-selection peptide sequences. The evolutionary filtering and sequence masking should also be switched off but redundancy in the peptides should be removed prior to analysis, unless it represents true independent enrichment.

Domain binding and phosphorylation targets obtained with these methods can also serve as input for specialized SLiM discovery tools.

MOTIPS [134] converts the given data into a sequence profile (after a normalization step to ensure consistent scoring among evidence from different sources), which is used in turn for whole-proteome scans that produce a list of potential domain targets. The score for each putative motif is combined with feature assessments based on residue-specific, precomputed values of conservation, solvent accessibility and disorder bias and then compared with a validated sequence set. The final output of MOTIPS is a ranked list of motif hits according to the likelihood of interacting with the domain of interest. In this fashion, the ability to independently recover the domain used for the original experiment can be used to assess the success or failure in motif prediction.

3.7 Challenges to Interpretation of De Novo SLiM Predictions

Edwards et al. [63] provide a fairly detailed discussion of the challenges in interpreting de novo SLiM predictions. Fundamentally, there are two connected questions:

1. Is the motif genuinely enriched? (i.e., Is the statistical model good?)
2. Is the enrichment for the reasons postulated when the dataset was constructed?

Both questions are arguably impossible to answer by bioinformatics alone, although robust benchmarking and data exploration can get a good handle on the former. Assuming that the SLiM prediction program functions as intended, the question then becomes whether the assumptions of the statistical model are valid or whether violations of that model could generate false positive enrichment. Again, the big consideration here is one of underlying protein sequence bias versus motif-specific sequence bias. Trying different ways of masking the data and/or generating the background model can help get a handle on this. It is also important to check thoroughly for evolutionary relationships between sequences that may have escaped detection.

The second question is usually more important but harder to get a handle on: given that a set of target sequences S were selected for a reason (e.g., common PPI partner or subcellular location), and the analysis show they are enriched for motif M, what is the causal relationship between M and S? Essentially three explanations are feasible:

1. M is (at least in part) responsible for S. This is usually the desired outcome, and therefore the default explanation, but it should be concluded with caution without additional supporting data and/or follow-up experiments because the alternative explanations are also possible.
2. M is correlated with S but not causal. Non-independence of biological data makes it challenging to differentiate causation from correlation. Suppose, for example, all proteins in S bind a

protein A. The interactome of A is likely to be enriched for proteins targeted to a specific subcellular component C and/or share interactions with another protein B. Does motif M interact with protein A or protein B or target proteins to C?

3. M and S are unrelated. M is an enriched feature of the parent sequence dataset (e.g., the whole proteome) and its enrichment in S is purely chance.

There is currently no good way to distinguish between these explanations from an analysis of a single protein dataset but hints can be achieved by additional analyses. For example, specifically looking for enrichment in other PPI or GO datasets could give hints regarding non-causal correlations, whilst analyzing randomly assembled datasets of real proteins from the same source can give insights into nonspecific enrichment. (*See* [11] and [63] for further discussion of these issues.) Correlating occurrences of predicted SLiMs with different biological features in a similar vein to FIRE-pro [124] might prove to be very helpful in this endeavor.

Flanking regions of SLiMs have been shown to be important for both function and specificity of binding [47, 98]. It is therefore not surprising that patterns returned from de novo SLiM prediction of known motifs (i.e., true positive benchmarks) frequently include flanking residues beyond the database definition. There could be several reasons for this. Chance enrichment of particular flanking residues could result in the longer SLiM being significantly enriched due to enrichment of the shorter versions. Alternatively, the flanking residues could belong to a second co-occurring motif as part of a "switch" in which two neighboring or overlapping SLiMs mediate mutually exclusive binding [13, 60], possibly by the steric hindrance introduced by the bound globular domain [8]. Finally, of course, there remains the possibility that literature/database definition of the SLiM is incomplete, and that the enriched flanking position is actually part of the SLiM or a sub-class thereof.

4 Concluding Remarks

Computational SLiM prediction is a blossoming field with new methods being developed on a regular basis. Whilst welcome, this can be confusing for the uninitiated, who may struggle to choose from the various tools available. There is no universal best solution to all SLiM prediction problems and so the nature of the input data as well as any potential follow up must be taken into consideration. Is the task motif instance prediction, or de novo discovery? Is the target of the search a single protein of interest, an alignment, a small dataset of multiple proteins, or a whole proteome/interactome? Are motifs likely to be shared by family members and thus have arisen by divergent evolution across homologues, or are they independent convergently evolved instances in unrelated proteins? Modelling the latter is the most common approach for de novo prediction but it is crucial to correct for evolutionary relationships in the data or else the former will be identified without realizing it. Despite the advances made by DILIMOT [71, 113], SLiMDisc [77] and SLiMFinder [41] in this area, a surprising number of de novo discovery tools are still published that overlook this fundamental discovery bias.

Performance benchmarking is often overlooked but this is vital if one is to understand the strengths, weaknesses and biases of the predictions produced. We have developed SLiMBench and made it part of the SLiMSuite package, which we hope will make this exercise easier in future. Not only can this enable direct comparisons of method performance, it can also help optimize parameter settings for SLiM discovery. The importance of an estimate of statistical significance for de novo predictions cannot be overstated. Is there really a motif to be found in the data? If there may not be, what is the False Positive Rate of the method being applied and what implications does this have given the scale of the analysis and planned experimental follow up? This is particularly crucial for large-scale analyses. Statistical significance is not only important for estimated False Discovery Rates; it is the only metric that is comparable between datasets with different sequence numbers, lengths, and/or composition.

Computational SLiM discovery has made a lot of progress over the last decade in successfully identifying overrepresented motifs. Nevertheless, Davey et al. point out that "Computational approaches, which should lead and focus experimental discovery, are in many ways lagging behind the advances of the experimentalists… [and] have yet to reveal the expected multitude of novel motif classes and instances." [1] Methods that differentiate between causal and coincidental enrichment are the key to the future success of bioinformatics approaches to this challenging yet important biological problem.

References

1. Davey NE, Van Roey K, Weatheritt RJ et al (2012) Attributes of short linear motifs. Mol Biosyst 8(1):268–281
2. Pawson T (1995) Protein modules and signalling networks. Nature 373(6515):573–580
3. Davis BD, Tai PC (1980) The mechanism of protein secretion across membranes. Nature 283(5746):433–438
4. Aasland R, Abrams C, Ampe C et al (2002) Normalization of nomenclature for peptide motifs as ligands of modular protein domains. FEBS Lett 513(1):141–144
5. Puntervoll P, Linding R, Gemund C et al (2003) ELM server: a new resource for investigating short functional sites in modular eukaryotic proteins. Nucleic Acids Res 31(13):3625–3630
6. Pancsa R, Fuxreiter M (2012) Interactions via intrinsically disordered regions: what kind of motifs? IUBMB Life 64(6):513–520

7. Neduva V, Russell RB (2006) Peptides mediating interaction networks: new leads at last. Curr Opin Biotechnol 17(5):465–471

8. Diella F, Haslam N, Chica C et al (2008) Understanding eukaryotic linear motifs and their role in cell signaling and regulation. Front Biosci 13:6580–6603

9. Dinkel H, Van Roey K, Michael S et al (2014) The eukaryotic linear motif resource ELM: 10 years and counting. Nucleic Acids Res 42(1):D259–D266

10. Mi T, Merlin JC, Deverasetty S et al (2012) Minimotif Miner 3.0: database expansion and significantly improved reduction of false-positive predictions from consensus sequences. Nucleic Acids Res 40(Database issue):D252–D260

11. Davey NE, Edwards RJ, Shields DC (2010) Computational identification and analysis of protein short linear motifs. Front Biosci (Landmark Ed) 15:801–825

12. Neduva V, Russell RB (2005) Linear motifs: evolutionary interaction switches. FEBS Lett 579(15):3342–3345

13. Van Roey K, Gibson TJ, Davey NE (2012) Motif switches: decision-making in cell regulation. Curr Opin Struct Biol 22(3):378–385

14. Vyas J, Nowling RJ, Maciejewski MW et al (2009) A proposed syntax for Minimotif Semantics, version 1. BMC Genomics 10:360

15. Davey NE, Trave G, Gibson TJ (2011) How viruses hijack cell regulation. Trends Biochem Sci 36(3):159–169

16. Garamszegi S, Franzosa EA, Xia Y (2013) Signatures of pleiotropy, economy and convergent evolution in a domain-resolved map of human-virus protein-protein interaction networks. PLoS Pathog 9(12):e1003778

17. Davey NE, Edwards RJ, Shields DC (2010) Estimation and efficient computation of the true probability of recurrence of short linear protein sequence motifs in unrelated proteins. BMC Bioinform 11:14

18. Sigrist CJ, Cerutti L, Hulo N et al (2002) PROSITE: a documented database using patterns and profiles as motif descriptors. Brief Bioinform 3(3):265–274

19. Xia X (2012) Position weight matrix, gibbs sampler, and the associated significance tests in motif characterization and prediction. Scientifica (Cairo) 2012:917540

20. Eddy SR (1998) Profile hidden Markov models. Bioinformatics 14(9):755–763

21. Krogh A, Brown M, Mian IS et al (1994) Hidden Markov models in computational biology. Applications to protein modeling. J Mol Biol 235(5):1501–1531

22. Obenauer JC, Cantley LC, Yaffe MB (2003) Scansite 2.0: proteome-wide prediction of cell signaling interactions using short sequence motifs. Nucleic Acids Res 31(13):3635–3641

23. Yoon BJ (2009) Hidden Markov models and their applications in biological sequence analysis. Curr Genomics 10(6):402–415

24. Seiler M, Mehrle A, Poustka A et al (2006) The 3of5 web application for complex and comprehensive pattern matching in protein sequences. BMC Bioinform 7:144

25. Davey NE, Haslam NJ, Shields DC et al (2010) SLiMSearch: a webserver for finding novel occurrences of short linear motifs in proteins, incorporating sequence context. Lect Notes Bioinform 6282:50–61

26. Meszaros B, Dosztanyi Z, Simon I (2012) Disordered binding regions and linear motifs–bridging the gap between two models of molecular recognition. PLoS One 7(10): e46829

27. Davey NE, Shields DC, Edwards RJ (2009) Masking residues using context-specific evolutionary conservation significantly improves short linear motif discovery. Bioinformatics 25(4):443–450

28. Brown CJ, Takayama S, Campen AM et al (2002) Evolutionary rate heterogeneity in proteins with long disordered regions. J Mol Evol 55(1):104–110

29. Tóth-Petróczy A, Mészáros B, Simon I et al (2008) Assessing conservation of disordered regions in proteins. Open Proteom J 1: 46–53

30. Fuxreiter M, Tompa P, Simon I (2007) Local structural disorder imparts plasticity on linear motifs. Bioinformatics 23(8):950–956

31. Remaut H, Waksman G (2006) Protein-protein interaction through beta-strand addition. Trends Biochem Sci 31(8):436–444

32. Cino EA, Choy WY, Karttunen M (2013) Conformational biases of linear motifs. J Phys Chem B 117(50):15943–15957

33. Abeln S, Frenkel D (2008) Disordered flanks prevent peptide aggregation. PLoS Comput Biol 4(12):e1000241

34. Sehnal D, Varekova RS, Huber HJ et al (2012) SiteBinder: an improved approach for comparing multiple protein structural motifs. J Chem Inf Model 52(2):343–359

35. Buljan M, Chalancon G, Eustermann S et al (2012) Tissue-specific splicing of disordered segments that embed binding motifs rewires protein interaction networks. Mol Cell 46(6):871–883

36. Weatheritt RJ, Davey NE, Gibson TJ (2012) Linear motifs confer functional diversity onto

splice variants. Nucleic Acids Res 40(15): 7123–7131

37. Weatheritt RJ, Gibson TJ (2012) Linear motifs: lost in (pre)translation. Trends Biochem Sci 37(8):333–341
38. Wan J, Qian SB (2014) TISdb: a database for alternative translation initiation in mammalian cells. Nucleic Acids Res 42(1):D845–D850
39. Kochetov AV (2008) Alternative translation start sites and hidden coding potential of eukaryotic mRNAs. Bioessays 30(7):683–691
40. UniProt C (2014) Activities at the universal protein resource (UniProt). Nucleic Acids Res 42(1):D191–D198
41. Edwards RJ, Davey NE, Shields DC (2007) SLiMFinder: a probabilistic method for identifying over-represented, convergently evolved, short linear motifs in proteins. PLoS One 2(10):e967
42. Davey NE, Edwards RJ, Shields DC (2007) The SLiMDisc server: short, linear motif discovery in proteins. Nucleic Acids Res 35(Web Server issue):W455–W459
43. Flicek P, Amode MR, Barrell D et al (2014) Ensembl 2014. Nucleic Acids Res 42(1):D749–D755
44. Oldfield CJ, Cheng Y, Cortese MS et al (2005) Coupled folding and binding with alpha-helix-forming molecular recognition elements. Biochemistry 44(37): 12454–12470
45. Mohan A, Oldfield CJ, Radivojac P et al (2006) Analysis of molecular recognition features (MoRFs). J Mol Biol 362(5):1043–1059
46. Vacic V, Oldfield CJ, Mohan A et al (2007) Characterization of molecular recognition features, MoRFs, and their binding partners. J Proteome Res 6(6):2351–2366
47. Stein A, Aloy P (2008) Contextual specificity in peptide-mediated protein interactions. PLoS One 3(7):e2524
48. Teyra J, Sidhu SS, Kim PM (2012) Elucidation of the binding preferences of peptide recognition modules: SH3 and PDZ domains. FEBS Lett 586(17):2631–2637
49. Liu Y, Woods NT, Kim D et al (2011) Yeast two-hybrid junk sequences contain selected linear motifs. Nucleic Acids Res 39(19):e128
50. Eisenhaber B, Eisenhaber F (2010) Prediction of posttranslational modification of proteins from their amino acid sequence. Methods Mol Biol 609:365–384
51. Trost B, Kusalik A (2011) Computational prediction of eukaryotic phosphorylation sites. Bioinformatics 27(21):2927–2935
52. Sigrist CJ, De Castro E, Langendijk-Genevaux PS et al (2005) ProRule: a new database containing functional and structural information on PROSITE profiles. Bioinformatics 21(21): 4060–4066
53. Sigrist CJ, de Castro E, Cerutti L et al (2013) New and continuing developments at PROSITE. Nucleic Acids Res 41(Database issue):D344–D347
54. Letunic I, Doerks T, Bork P (2012) SMART 7: recent updates to the protein domain annotation resource. Nucleic Acids Res 40(Database issue):D302–D305
55. Punta M, Coggill PC, Eberhardt RY et al (2012) The Pfam protein families database. Nucleic Acids Res 40(Database issue): D290–D301
56. Chica C, Labarga A, Gould CM et al (2008) A tree-based conservation scoring method for short linear motifs in multiple alignments of protein sequences. BMC Bioinform 9:229
57. Via A, Gould CM, Gemund C et al (2009) A structure filter for the Eukaryotic Linear Motif Resource. BMC Bioinform 10:351
58. Weatheritt RJ, Jehl P, Dinkel H et al (2012) iELM–a web server to explore short linear motif-mediated interactions. Nucleic Acids Res 40(Web Server issue):W364–W369
59. Dinkel H, Chica C, Via A et al (2011) Phospho.ELM: a database of phosphorylation sites–update 2011. Nucleic Acids Res 39(Database issue):D261–D267
60. Van Roey K, Dinkel H, Weatheritt RJ et al (2013) The switches.ELM resource: a compendium of conditional regulatory interaction interfaces. Sci Signal 6(269):rs7
61. Jin J, Pawson T (2012) Modular evolution of phosphorylation-based signalling systems. Philos Trans R Soc Lond B Biol Sci 367(1602):2540–2555
62. Songyang Z, Blechner S, Hoagland N et al (1994) Use of an oriented peptide library to determine the optimal substrates of protein kinases. Curr Biol 4(11):973–982
63. Edwards RJ, Davey NE, O'Brien K et al (2012) Interactome-wide prediction of short, disordered protein interaction motifs in humans. Mol Biosyst 8(1):282–295
64. Ashburner M, Ball CA, Blake JA et al (2000) Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. Nat Genet 25(1):25–29
65. Hamosh A, Scott AF, Amberger J et al (2000) Online mendelian inheritance in man (OMIM). Hum Mutat 15(1):57–61
66. Goel R, Harsha HC, Pandey A et al (2012) Human protein reference database and human proteinpedia as resources for phosphoproteome analysis. Mol Biosyst 8(2):453–463

67. Safran M, Dalah I, Alexander J et al (2010) GeneCards Version 3: the human gene integrator. Database (Oxford) 2010:baq020

68. Davey NE, Haslam NJ, Shields DC et al (2011) SLiMSearch 2.0: biological context for short linear motifs in proteins. Nucleic Acids Res 39(Web Server issue):W56–W60

69. Edwards RJ, Davey NE, Shields DC (2008) CompariMotif: quick and easy comparisons of sequence motifs. Bioinformatics 24(10): 1307–1309

70. Marsico A, Scheubert K, Tuukkanen A et al (2010) MeMotif: a database of linear motifs in alpha-helical transmembrane proteins. Nucleic Acids Res 38(Database issue):D181–D189

71. Neduva V, Linding R, Su-Angrand I et al (2005) Systematic discovery of new recognition peptides mediating protein interaction networks. PLoS Biol 3(12):e405

72. Bailey TL, Boden M, Buske FA et al (2009) MEME SUITE: tools for motif discovery and searching. Nucleic Acids Res 37(Web Server issue):W202–W208

73. Grant CE, Bailey TL, Noble WS (2011) FIMO: scanning for occurrences of a given motif. Bioinformatics 27(7):1017–1018

74. Frith MC, Saunders NF, Kobe B et al (2008) Discovering sequence motifs with arbitrary insertions and deletions. PLoS Comput Biol 4(4):e1000071

75. Bailey TL, Gribskov M (1997) Score distributions for simultaneous matching to multiple motifs. J Comput Biol 4(1):45–59

76. de Castro E, Sigrist CJ, Gattiker A et al (2006) ScanProsite: detection of PROSITE signature matches and ProRule-associated functional and structural residues in proteins. Nucleic Acids Res 34(Web Server issue):W362–W365

77. Davey NE, Shields DC, Edwards RJ (2006) SLiMDisc: short, linear motif discovery, correcting for common evolutionary descent. Nucleic Acids Res 34(12):3546–3554

78. Peng ZL, Kurgan L (2012) Comprehensive comparative assessment of in-silico predictors of disordered regions. Curr Protein Pept Sci 13(1):6–18

79. Deng X, Eickholt J, Cheng J (2012) A comprehensive overview of computational protein disorder prediction methods. Mol Biosyst 8(1):114–121

80. Dosztanyi Z, Csizmok V, Tompa P et al (2005) IUPred: web server for the prediction of intrinsically unstructured regions of proteins based on estimated energy content. Bioinformatics 21(16):3433–3434

81. Haslam NJ, Shields DC (2012) Profile-based short linear protein motif discovery. BMC Bioinform 13:104

82. Sickmeier M, Hamilton JA, LeGall T et al (2007) DisProt: the database of disordered proteins. Nucleic Acids Res 35(Database issue):D786–D793

83. Chen JW, Romero P, Uversky VN et al (2006) Conservation of intrinsic disorder in protein domains and families: I. A database of conserved predicted disordered regions. J Proteome Res 5(4):879–887

84. Tompa P, Fuxreiter M, Oldfield CJ et al (2009) Close encounters of the third kind: disordered domains and the interactions of proteins. Bioessays 31(3):328–335

85. Williams RW, Xue B, Uversky VN et al (2013) Distribution and cluster analysis of predicted intrinsically disordered protein Pfam domains. Intrins Disord Prot 1:e25724

86. Schaeffer RD, Jonsson AL, Simms AM et al (2011) Generation of a consensus protein domain dictionary. Bioinformatics 27(1):46–54

87. Towse CL, Daggett V (2012) When a domain is not a domain, and why it is important to properly filter proteins in databases: conflicting definitions and fold classification systems for structural domains make filtering of such databases imperative. Bioessays 34(12):1060–1069

88. Linding R, Russell RB, Neduva V et al (2003) GlobPlot: exploring protein sequences for globularity and disorder. Nucleic Acids Res 31(13):3701–3708

89. Mosca R, Ceol A, Stein A et al (2014) 3did: a catalog of domain-based interactions of known three-dimensional structure. Nucleic Acids Res 42(1):D374–D379

90. Stein A, Aloy P (2010) Novel peptide-mediated interactions derived from high-resolution 3-dimensional structures. PLoS Comput Biol 6(5):e1000789

91. Brannetti B, Helmer-Citterich M (2003) iSPOT: a web tool to infer the interaction specificity of families of protein modules. Nucleic Acids Res 31(13):3709–3711

92. Trabuco LG, Lise S, Petsalaki E et al (2012) PepSite: prediction of peptide-binding sites from protein surfaces. Nucleic Acids Res 40(Web Server issue):W423–W427

93. Perrodou E, Chica C, Poch O et al (2008) A new protein linear motif benchmark for multiple sequence alignment software. BMC Bioinform 9:213

94. Sayers EW, Barrett T, Benson DA et al (2011) Database resources of the National Center for

Biotechnology Information. Nucleic Acids Res 39(Database issue):D38–D51

95. Balla S, Thapar V, Verma S et al (2006) Minimotif Miner: a tool for investigating protein function. Nat Methods 3(3):175–177
96. Dinkel H, Sticht H (2007) A computational strategy for the prediction of functional linear peptide motifs in proteins. Bioinformatics 23(24):3297–3303
97. Davey NE, Cowan JL, Shields DC et al (2012) SLiMPrints: conservation-based discovery of functional motif fingerprints in intrinsically disordered protein regions. Nucleic Acids Res 40(21):10628–10641
98. Chica C, Diella F, Gibson TJ (2009) Evidence for the concerted evolution between short linear protein motifs and their flanking regions. PLoS One 4(7):e6052
99. O'Brien KT, Haslam NJ, Shields DC (2013) SLiMScape: a protein short linear motif analysis plugin for Cytoscape. BMC Bioinform 14:224
100. Davey NE, Haslam NJ, Shields DC et al (2010) SLiMFinder: a web server to find novel, significantly over-represented, short protein motifs. Nucleic Acids Res 38(Web Server issue):W534–W539
101. Plewczynski D, Basu S, Saha I (2012) AMS 4.0: consensus prediction of post-translational modifications in protein sequences. Amino Acids 43(2):573–582
102. Kerrien S, Aranda B, Breuza L et al (2012) The IntAct molecular interaction database in 2012. Nucleic Acids Res 40(Database issue):D841–D846
103. Altschul SF, Madden TL, Schaffer AA et al (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids Res 25(17):3389–3402
104. Via A, Gherardini PF, Ferraro E et al (2007) False occurrences of functional motifs in protein sequences highlight evolutionary constraints. BMC Bioinform 8:68
105. Nguyen Ba AN, Yeh BJ, van Dyk D et al (2012) Proteome-wide discovery of evolutionary conserved sequences in disordered regions. Sci Signal 5(215):rs1
106. Fang C, Noguchi T, Tominaga D et al (2013) MFSPSSMpred: identifying short disorder-to-order binding regions in disordered proteins based on contextual local evolutionary conservation. BMC Bioinform 14:300
107. Chou MF, Schwartz D (2011) Biological sequence motif discovery using motif-x. *Curr Protoc Bioinform* Chapter 13, Unit 13 15–24
108. Orchard S (2012) Molecular interaction databases. Proteomics 12(10):1656–1662
109. Jonassen I (1997) Efficient discovery of conserved patterns using a pattern graph. Comput Appl Biosci 13(5):509–522
110. Jonassen I, Collins JF, Higgins DG (1995) Finding flexible patterns in unaligned protein sequences. Protein Sci 4(8):1587–1595
111. Neuwald AF, Green P (1994) Detecting patterns in protein sequences. J Mol Biol 239(5):698–712
112. Rigoutsos I, Floratos A (1998) Combinatorial pattern discovery in biological sequences: the TEIRESIAS algorithm. Bioinformatics 14(1):55–67
113. Neduva V, Russell RB (2006) DILIMOT: discovery of linear motifs in proteins. Nucleic Acids Res 34(Web Server issue): W350–W355
114. Bailey TL, Elkan C (1994) Fitting a mixture model by expectation maximization to discover motifs in biopolymers. Proc Int Conf Intell Syst Mol Biol 2:28–36
115. Lawrence CE, Reilly AA (1990) An expectation maximization (EM) algorithm for the identification and characterization of common sites in unaligned biopolymer sequences. Proteins 7(1):41–51
116. Do CB, Batzoglou S (2008) What is the expectation maximization algorithm? Nat Biotechnol 26(8):897–899
117. Down TA, Hubbard TJ (2005) NestedMICA: sensitive inference of over-represented motifs in nucleic acid sequence. Nucleic Acids Res 33(5):1445–1453
118. Dogruel M, Down TA, Hubbard TJ (2008) NestedMICA as an ab initio protein motif discovery tool. BMC Bioinform 9:19
119. Dinh H, Rajasekaran S (2013) PMS: a panoptic motif search tool. PLoS One 8(12):e80660
120. Dinh H, Rajasekaran S, Davila J (2012) qPMS7: a fast algorithm for finding (l, d)-motifs in DNA and protein sequences. PLoS One 7(7):e41425
121. Tan SH, Hugo W, Sung WK et al (2006) A correlated motif approach for finding short linear motifs from protein interaction networks. BMC Bioinform 7:502
122. Leung HC, Siu MH, Yiu SM et al (2009) Clustering-based approach for predicting motif pairs from protein interaction data. J Bioinform Comput Biol 7(4):701–716
123. Boyen P, Van Dyck D, Neven F et al (2011) SLIDER: a generic metaheuristic for the discovery of correlated motifs in protein-protein interaction networks. IEEE/ACM Trans Comput Biol Bioinform 8(5):1344–1357
124. Lieber DS, Elemento O, Tavazoie S (2010) Large-scale discovery and characterization of

protein regulatory motifs in eukaryotes. PLoS One 5(12):e14444

125. Dosztanyi Z, Meszaros B, Simon I (2009) ANCHOR: web server for predicting protein binding regions in disordered proteins. Bioinformatics 25(20):2745–2746
126. Meszaros B, Simon I, Dosztanyi Z (2009) Prediction of protein binding regions in disordered proteins. PLoS Comput Biol 5(5):e1000376
127. Cheng Y, Oldfield CJ, Meng J et al (2007) Mining alpha-helix-forming molecular recognition features with cross species sequence alignments. Biochemistry 46(47): 13468–13477
128. Disfani FM, Hsu WL, Mizianty MJ et al (2012) MoRFpred, a computational tool for sequence-based prediction and characterization of short disorder-to-order transitioning binding regions in proteins. Bioinformatics 28(12):i75–i83
129. Mooney C, Pollastri G, Shields DC et al (2012) Prediction of short linear protein binding regions. J Mol Biol 415(1):193–204
130. Rose PW, Bi C, Bluhm WF et al (2013) The RCSB Protein Data Bank: new resources for research and education. Nucleic Acids Res 41(Database issue):D475–D482
131. Betel D, Breitkreuz KE, Isserlin R et al (2007) Structure-templated predictions of novel protein interactions from sequence information. PLoS Comput Biol 3(9):1783–1789
132. Hugo W, Sung WK, Ng SK (2013) Discovering interacting domains and motifs in protein-protein interactions. Methods Mol Biol 939:9–20
133. Gibson TJ (2009) Cell regulation: determined to signal discrete cooperation. Trends Biochem Sci 34(10):471–482
134. Lam HY, Kim PM, Mok J et al (2010) MOTIPS: automated motif analysis for predicting targets of modular protein domains. BMC Bioinform 11:243
135. Schwartz D, Gygi SP (2005) An iterative statistical approach to the identification of protein phosphorylation motifs from large-scale data sets. Nat Biotechnol 23(11): 1391–1398
136. Schwartz D, Chou MF, Church GM (2009) Predicting protein post-translational modifications using meta-analysis of proteome scale data sets. Mol Cell Proteomics 8(2): 365–379
137. Villen J, Beausoleil SA, Gerber SA et al (2007) Large-scale phosphorylation analysis of mouse liver. Proc Natl Acad Sci U S A 104(5): 1488–1493
138. Wilson-Grady JT, Villen J, Gygi SP (2008) Phosphoproteome analysis of fission yeast. J Proteome Res 7(3):1088–1097
139. Zhai B, Villen J, Beausoleil SA et al (2008) Phosphoproteome analysis of Drosophila melanogaster embryos. J Proteome Res 7(4):1675–1682
140. Edwards RJ. SLiMSuite software package. 2013 [cited 25/1/14]; Available from: http://www.southampton.ac.uk/~re1u06/software/packages/slimsuite/