

Systematic and Fully Automated Identification of Protein Sequence Patterns

REECE K. HART,^{1,2} AJAY K. ROYYURU,¹ GUSTAVO STOLOVITZKY,¹
and ANDREA CALIFANO^{1,3}

ABSTRACT

We present an efficient algorithm to systematically and automatically identify patterns in protein sequence families. The procedure is based on the Splash deterministic pattern discovery algorithm and on a framework to assess the statistical significance of patterns. We demonstrate its application to the fully automated discovery of patterns in 974 PROSITE families (the complete subset of PROSITE families which are defined by patterns and contain DR records). Splash generates patterns with better specificity and undiminished sensitivity, or vice versa, in 28% of the families; identical statistics were obtained in 48% of the families, worse statistics in 15%, and mixed behavior in the remaining 9%. In about 75% of the cases, Splash patterns identify sequence sites that overlap more than 50% with the corresponding PROSITE pattern. The procedure is sufficiently rapid to enable its use for daily curation of existing motif and profile databases. Third, our results show that the statistical significance of discovered patterns correlates well with their biological significance. The trypsin subfamily of serine proteases is used to illustrate this method's ability to exhaustively discover all motifs in a family that are statistically and biologically significant. Finally, we discuss applications of sequence patterns to multiple sequence alignment and the training of more sensitive score-based motif models, akin to the procedure used by PSI-BLAST. All results are available at <http://www.research.ibm.com/spat/>.

Key words: pattern discovery, PROSITE, motif, profile, hidden Markov model.

1. INTRODUCTION

THE RAPID ADVANCEMENTS IN SEQUENCING TECHNOLOGIES and exponential growth in genomic databases are spurring the development of techniques for the identification of sequence motifs and sequence classification. This is commonly accomplished by defining sequence signatures that distinguish a family or set of sequences from the complete sequence database which facilitates the classification of new sequences into these families (Bork *et al.*, 1996). Sequence signatures may be defined by simple consensus or regular expression patterns, often called sequence motifs, or by more elaborate scoring methods, such as position specific scoring matrices (PSSMs) (Henikoff *et al.*, 1999), profiles (Gribnikov *et al.*, 1987), and Hidden Markov Models (HMMs) (Durbin *et al.*, 1998). Whereas a motif either does or does not match a sequence,

¹IBM Computational Biology Center, T.J. Watson Research Center, PO Box 218, Yorktown Heights, NY 10598.

²Permanent address: <http://www.in-machina.com/reece/>, reece@in-machina.com

³Present address: First Genetics Trust.

scoring methods quantitate the degree of match. Either approach is capable of treating variable length gaps in protein sequences. Although the scoring methodologies are generally believed to provide more sensitive classification, sequence motifs remain attractive because of the relative ease of specification and use. In practice, sequence motifs are often used to bootstrap the training of score-based methods.

There exist several well-curated and established compilations of sequence signatures, such as PROSITE (Bairoch, 1991), ProDom (Sonnhammer *et al.*, 1994), PRINTS (Attwood *et al.*, 1999), PFAM (Bateman *et al.*, 1999), and BLOCKS (Henikoff *et al.*, 1999). The databases differ in the methods used to generate sequence signatures, whether these signatures are used individually or jointly, whether gaps are accommodated, and whether matches are scored or are Boolean. Several of these databases are now available through InterPro, the integrated documentation resource for protein families, domains, and functional sites (<http://www.ebi.ac.uk/interpro/>). The curation of sequence signature databases is a labor-intensive task and is increasingly challenged by the rapid explosion of data in genomic repositories. Attempts to automate the process have resulted in several new methodologies (see Brazma (1998) for a review) for the identification of conserved sequence motifs, such as MEME (Bailey *et al.*, 1994), the Gibbs Sampler (Neuwald *et al.*, 1995), Pratt (Jonassen, 1997), EMOTIF (Nevill-Manning *et al.*, 1998), Teiresias (Rigoutsos *et al.*, 1998), and Splash (Califano, 2000).

Among these databases, PROSITE is especially relevant because of the high biological significance of the reported patterns. The PROSITE database version 15.0 contains extensively annotated collections of 1352 motifs or profiles grouped into 1014 protein families. Each PROSITE entry stems from a set of protein sequences grouped by an expert, using biological information which is provided as documentation. For almost all entries, PROSITE provides a sequence motif that characterizes the functionally relevant residues of a protein family. These are obtained by selecting regions of sequences that have a documented functional significance and by performing multiple sequence alignment over these selected regions to identify consensus patterns. Beyond the inherent simplicity and utility of the sequence patterns, PROSITE also serves as a very useful database of sequence classifications to guide the development of derivative databases based on other methodologies.

Here we report on the systematic application and evaluation of Splash, a deterministic pattern-discovery algorithm, in combination with a framework for the analysis of the statistical significance of patterns (Stolovitzky *et al.*, 1999) and demonstrate its application to the identification of highly conserved motifs in protein families. Although others have investigated the importance of a statistical underpinning to pattern discovery in selected families (see, for example, Neuwald *et al.*, 1994), we instead chose to apply our procedure to all 974 PROSITE PDOC families associated with one or more motifs. We present the overall sensitivity and specificity of the automatically discovered patterns relative to PROSITE and show the strong correlation between these performance measures and the statistical significance of patterns. By assessing the extent of overlap between PROSITE and Splash-generated patterns, we demonstrate that Splash correctly infers *biologically* significant regions in most cases. This analysis is extended to illustrate instances where our procedure suggests refinements of the biologically relevant PROSITE patterns that improve their sensitivity and/or specificity. Some examples in which PROSITE patterns perform better than Splash patterns will be discussed. Finally, we report on the exhaustive discovery and analysis of the motifs that occur in at least 40% of the sequences in the trypsin family and compare these with motifs reported in the PROSITE and BLOCKS databases.

Due to the high quality of the annotations in PROSITE, it is useful to systematically and objectively compare the results of any new technique with those reported by PROSITE, in terms both of sensitivity and specificity (false negatives and false positives) and of the ability to identify regions that are biologically

TABLE 1. SUBSTITUTION SETS FOR THE BLOSUM50 SUBSTITUTION MATRIX WITH THRESHOLD $m_0 = 0^1$

H(A)={A, S}	H(F)={F, L, W, Y}	H(M)={I, L, M, V}	H(S)={A, N, S, T}
H(B)={D, E, N}	H(G)={G}	H(N)={D, H, N, S}	H(T)={S, T}
H(C)={C}	H(H)={H, N, Q, Y}	H(P)={P}	H(V)={I, L, M, V}
H(D)={D, E, N}	H(I)={I, L, M, V}	H(Q)={E, H, K, Q, R}	H(W)={F, W, Y}
H(E)={D, E, K, Q}	H(L)={F, I, L, M, V}	H(R)={K, Q, R}	H(Y)={F, H, W, Y}
			H(Z)={D, E, K, Q}

¹For a matrix \mathbf{m} and alphabet of amino acids \mathcal{A} , the substitution set $H(\mathbf{a})$ is given by $H(\mathbf{a}) = \{a_s | \mathbf{m}(a, a_s) > m_0\} \forall a \in \mathcal{A}$.

significant (Henikoff *et al.*, 1991). It is in this spirit that we have chosen to compare the results of our fully automated pattern discovery protocol to the results in PROSITE. Once such patterns have been identified, they may be used directly or to seed the training of PSSM or HMM methods.

This investigation demonstrates the utility of automated pattern-discovery methods for the maintenance of current motif databases. We emphasize that the entire process is automated and performed identically for all PROSITE families; we have made no effort to tune parameters for specific families. Although we have used PROSITE families for analysis and comparison, we have not made use of the pattern information contained in a PROSITE record in the automated pattern-discovery process.

2. METHODS

2.1. Notation and definitions

We use *perl*-style regular expression syntax (Wall *et al.*, 1996) to describe a pattern in protein sequences. A pattern is a sequence of tokens that may be a single amino acid (e.g., **G** for glycine), a set of amino acids (e.g., **[AG]** for one residue, alanine *or* glycine), or a wildcard which allows any *single* amino acid (denoted by **.**). Tokens which are not wildcard characters are called *solid*. Tokens which specify two or more amino acids are called *similarity tokens* and form a *similarity set*. If tokens in a similarity set are a subset of those consistent with a specified substitution matrix and threshold, the tokens form a *substitution set*; otherwise, they are said to be *promiscuous*. Table 1 enumerates the substitution sets for the BLOSUM50 (Henikoff *et al.*, 1992) matrix and substitution threshold of 0.0. For example, **[^DN]** is a valid substitution token and **[^DNR]** is a promiscuous token with respect to BLOSUM50 and threshold 0. A pattern which contains a similarity set is a *similarity pattern*. In a similarity pattern, if any similarity set is promiscuous, the pattern itself is promiscuous; otherwise, the pattern is simply a *substitution pattern*.

Any token may be optionally followed by a repeat count of the form $\{a, b\}$ which requires at least a and no more than b contiguous instances of that token; in the absence of specification, exactly one instance is assumed. The *length* of a pattern is the number of solid tokens in the pattern, whereas the *span* is the total number of tokens. Patterns which have a fixed span are *rigid*, while those that do not (i.e., have one or more variable length repeats) are termed *flexible*. The number of solids in a specified window size can be interpreted as the *token density*. Because token density influences pattern specificity and computational workload, we use a *density constraint* which imposes a minimum token density in the generated patterns.

Patterns are discovered on a set of protein sequences, called the *target set*. The target set is typically a subset of a much larger *universe* of protein sequences, such as SWISS-PROT. We refer to the number of sequences in the target set which are matched by a pattern as the *support* for that pattern. The resulting set of patterns form the *discovery set*. The *sensitivity* of a pattern is measured by the number of false negatives, i.e., the number of sequences in the target set which are *not* matched by the pattern. The *specificity* is measured by the number of false positives, i.e., the number of sequences in the universe which are matched by the pattern but do not belong to the target set.

2.2. Pattern scoring

The utility of a pattern is a mixture of its specificity and sensitivity. Determining these directly requires searching the target set and universe of sequences respectively, and doing so for a set of patterns is time consuming. We use the following equations to estimate the likelihood of observing a particular pattern using only the composition of the pattern and amino acid frequencies.

Patterns with k tokens and l total characters are called kl -patterns. Those kl -patterns with support j are called jkl -patterns. Given a kl -pattern π , the probability that it occurs at least once in a sequence of length L is $1 - (1 - \rho_\pi)^{L-1}$, where

$$\rho_\pi = \prod_{i=1 \dots k} \wp(v_i) \quad (2.1)$$

is the probability of pattern π with the k tokens to occur in a sequence and

$$\wp(v) \cong \sum_{a \in v} f(a) \quad (2.2)$$

is the probability of a similarity token ν to occur in sequence at a given position, and $f(a)$ is the frequency with which amino acid a occurs in the universe of protein sequences. For example, $\wp(\text{ILMV}) \cong f(\text{I}) + f(\text{L}) + f(\text{M}) + f(\text{V})$. The approximation consists of neglecting the overlaps between substitution tokens. Equation 2.2 will overestimate $\wp(\nu)$, and this will result in an overestimation of the expected number of patterns (see below).

As shown previously (Stolovitzky *et al.*, 1999), the average number of maximal jkl -patterns that satisfies the $\langle k, l \rangle$ density constraint and appearing in a random database composed of n sequences of length L , is given by

$$\langle n_{jkl} \rangle = N_0(k, l) \binom{n}{j} \left\langle \frac{p^j (1-p)^{n-j}}{\rho_\pi} \langle p_{in} \rangle \langle p_{out} \rangle^2 \right\rangle. \quad (2.3)$$

The outer angular brackets refer to an average with respect to the matching probability of a generic pattern π , and $N_0(k, l)$ is the number of kl -patterns that satisfy the density constraint. Also, p_{in} and p_{out} are, respectively, the probability that a given jkl -pattern is maximal in composition and length for pattern π (Stolovitzky *et al.*, 1999). From this analysis, it is possible to estimate the probability of the number of discovered patterns that would have occurred in a random database of similar size and composition. This probability conforms approximately to a Poisson distribution and as such its mean and variance are approximately equal. Therefore, it is possible to compute a Z-score using only the above result as:

$$Z = \frac{n_{jkl} - \langle n_{jkl} \rangle}{\sigma_{n_{jkl}}} \quad (2.4)$$

where n_{jkl} is the number of discovered jkl -patterns. Details of this analysis, which is in excellent agreement with experimental data, are available in Califano (2000).

2.3. The Splash algorithm for pattern discovery

The Splash algorithm has been described in detail elsewhere (Califano, 2000), but will be briefly reviewed here. Splash deterministically discovers all rigid patterns in a set of unaligned input sequences subject to constraints of minimum support and token density. Constraining the support is equivalent to limiting the number of false negatives, i.e., to requiring *sensitive* patterns. The token density constraint limits the number of wildcard positions within a specified window size, but the overall length and span of the resulting pattern are not restricted. If an amino acid substitution matrix and threshold are provided, Splash will discover substitution patterns whose tokens are sets of amino acids within a substitution set. Substitution sets for the BLOSUM50 substitution matrix and matrix threshold of 0.0 are shown in Table 1. Each pattern generated by Splash is maximal in both 1) composition and 2) length. This means that a pattern cannot be made more specific by (1) replacing wildcards with solid tokens within the pattern or (2) adding solid tokens to the “left” or “right” of the pattern, without reducing the pattern sensitivity.

The inputs to the Splash algorithm are the unaligned sequences in the target set, the density constraint (specified as k tokens in a window l and denoted as $\langle k, l \rangle$), the minimum support required for a pattern (specified as a percentage of the number of sequences in the target set), and the substitution matrix and threshold for substitution patterns (Califano, 2000). The output from the algorithm is a set of all substitution patterns and associated Z-scores which meet the criteria specified by the parameters. The output is not dependent on the order of the input. Run times for the PROSITE families studied herein are typically 0.1 to 5 minutes on one CPU of a 266 MHz Pentium II-based computer.

2.4. Single pattern discovery in protein sequence families

Patterns in protein sequences are extremely varied in the density of tokens, number of identity tokens, the number of similarity tokens, the promiscuity of similarity tokens, and so forth. Accordingly, the parameters used to discover patterns in sequence databases depends on the diversity of the family being investigated. We describe an iterative use of the Splash algorithm to identify patterns which have high support and which are statistically significant (i.e., occur more frequently than expected in a random database of similar amino acid composition). We emphasize that this procedure was used without further tuning for all of the PROSITE families studied and reported herein.

Pattern discovery in each PROSITE family was performed by iteratively invoking the Splash algorithm while adjusting both the density constraint and the minimum support until at least n_0 patterns were reported. For each PROSITE pattern family, the target set of sequences was assembled from the true positive and false negative entries in the PROSITE pattern record. An initial density constraint $\langle k, l_{min} \rangle$ and minimum support j (equal to 100% of the sequences in the target set) were chosen. If fewer than n_0 patterns were reported, the density constraint was made less stringent by increasing l . If $l > l_{max}$ without discovering at least n_0 patterns, the minimum support j was decreased, l was reset to l_{min} , and the procedure was repeated. We do not change k . We require at least n_0 patterns in order to provide variability in the patterns and sequence regions which are covered. If a predefined support threshold j_{min} was reached without any pattern being discovered, the procedure is halted and no pattern was reported. As shown in Results, a study of the statistics of PROSITE 15.0 patterns suggested the parameters $k = 4$, $l_{min} = 8$, $l_{max} = 32$, and $j_{min} = 2$. As a final post-processing step, substitution tokens are “narrowed” to contain only the amino acids which are observed to occur at that position. For example, the pattern G[ILMV]G would be narrowed to G[IL]G if the patterns GMG and GVG were not observed in the target set. This procedure is depicted in Figure 1.

The deterministic nature of Splash guarantees that all patterns discovered using a set of density and minimum support parameters will also be reported for all lesser densities and support parameters. Because the pattern formation process is combinatorial, underestimation of density and minimum support parameters may cause a large number of closely related patterns to be generated in addition to ones which are more stringent. For example, if the pattern C.C has high support, the patterns AC.C, CAC, and C.CA are likely to occur as well, as are subsequent derivatives of those patterns. As a result, it is advisable to begin with high support and density requirements and then gradually relax this stringency as outlined above in order to limit the computational burden. The iterative procedure presented in this paper is designed to find patterns with the greatest support and statistical significance in a computationally efficient manner.

PROSITE 15.0 contains 1352 entries of patterns, rules, and profiles for 1014 protein families. For this study, we selected the 1,281 pattern records for which statistics (NR field of a PROSITE record) and sequences (DR field) are available. These patterns correspond to 974 PDOC families. In cases where there is more than one pattern for a PDOC family, we compare against only one pattern in that family.

2.5. Characterizing relative sensitivity, specificity, and overlap of PROSITE and Splash patterns

Splash may report a large number of patterns for each family. To limit the number of patterns evaluated, we chose the top 20 patterns by support and any other patterns with Z-scores greater than the maximum Z-score in the top 20. This selects the top 20 most sensitive patterns, as well as those which are expected to be more specific than these. These patterns form the discovery set. For each one of the Splash patterns in the discovery set, the number of false positives (n_{fp}^S) and false negatives (n_{fn}^S) are computed using criteria identical to that of PROSITE 15.0. Using the corresponding statistics n_{fp}^P and n_{fn}^P from the PROSITE record, the relative sensitivity $\Delta n_{fn} = n_{fn}^S - n_{fn}^P$ and relative specificity $\Delta n_{fp} = n_{fp}^S - n_{fp}^P$ are computed.

```

load target set
k=2
do j = 1.00N, 0.95N, ..., 2
  do l = 8, 16, 24, ...,
    run Splash(j,k,l)
    if found at least  $n_0$  patterns
      report best pattern
      if reported R patterns
        exit
    fi
  mask sequences
fi
od
od

```

FIG. 1. Pseudocode for the Single and Exhaustive Pattern Discovery protocols. The parameters j , k , and l are set and incremented/decremented as discussed in Section 2.4. Exhaustive pattern discovery is discussed in Section 2.6. The “best” pattern is determined by Z-score (Eqn. 2.4).

Negative, zero, and positive values correspond to increased, equal, and decreased sensitivity or specificity of a Splash pattern with respect to a PROSITE pattern. We define a penalty score (lower scores are better) that is used to compare the performance of different patterns:

$$S_p = \alpha \Delta n_{fn} + (1 - \alpha) \Delta n_{fp}. \quad (2.5)$$

Here, α is a positive constant used to weight the relative importance of false negatives and false positives. For instance, $\alpha = 1$ biases the score for sensitivity regardless of its specificity and $\alpha = 0.5$ gives a metric which weights false positive and false negative errors equally.

To quantify the extent to which patterns identify similar regions of sequence, the overlap between a PROSITE pattern, P , and Splash pattern, S , is computed as follows. The subset of true positive sequences in which both P and S match is identified. For each sequence in this subset, the loci $\{p_1, p_2, \dots, p_M\}$ and $\{s_1, s_2, \dots, s_N\}$ at which P and S match on the sequence is computed. Then, all pairwise signed relative offsets of P and S are evaluated as $\delta_{m,n} = s_n - p_m$ ($1 \leq m \leq M, 1 \leq n \leq N$). A set of all offsets for all sequences is histogrammed and the most frequently occurring δ is obtained from the histogram. In our results, this choice was obvious, unique, and supported by at least 50% of the offsets. The overlap o_{ps} is defined by

$$o_{ps} = \frac{|0, l_P| \cap |\delta, \delta + l_S|}{l_P} \quad (2.6)$$

where l_P and l_S are, the total number of characters, including tokens and wildcards, in P and S , respectively. If $o_{ps} = 1$, then S occurs within the boundaries of P ; if $o_{ps} = 0$, then the two patterns are incident on different regions of the sequences.

2.6. Exhaustive pattern discovery

More than one conserved nonoverlapping pattern can be discovered as follows. The procedure for Single Pattern Discovery is applied until at least n_0 patterns are discovered. The pattern with the lowest penalty score S_p is selected and reported. All occurrences of that pattern in the target set are masked (replaced by non-amino acid characters) so its tokens can no longer form other patterns. The procedure is resumed with density and support equal to those of the last run until another set of at least n_0 patterns is discovered. This process is repeated until the minimum support drops below a predefined threshold j_{min} . This procedure will find all the statistically significant disjoint motifs, each with support at least j_{min} , in the input. An application of this approach to the trypsin subfamily of serine proteases is presented in Results.

In order to assess the efficacy of this procedure to locate all known biologically relevant regions, we applied exhaustive pattern-discovery procedure to families which contain multiple PROSITE patterns. There are 266 such PDOC families in PROSITE. We discarded those families for which the union of true positives and false negatives sequences identified by each PROSITE pattern was not identical for all members of that PDOC family.

3. RESULTS

We first report on the statistical analysis of certain properties of the PROSITE patterns, such as their density and flexibility, and use these data to calibrate automated pattern discovery. In the subsequent sections, we present results of automated pattern discovery in increasing detail. We describe the overall sensitivity and specificity of the automatically generated patterns compared to those in PROSITE. We then show that our statistical framework performs well in ranking a set of patterns for sensitivity and specificity. To support the assertion that automatically generated patterns often discover regions known to be biologically relevant, we present results which compare the extent of overlap between PROSITE and Splash patterns on target sequences. We apply exhaustive pattern discovery to the trypsin family of proteases to show that pattern discovery in conjunction with the statistical framework efficiently identifies conserved regions of a protein family, including all three residues of the catalytic triad of the serine proteases. Finally, we assess our ability to locate all known biologically relevant regions in proteins by a systematic application of exhaustive pattern discovery to 98 families that contain multiple PROSITE patterns.

Terms appearing in italics are defined in Methods.

3.1. Statistics of patterns in PROSITE

In Figure 2, we report on the overall flexibility of PROSITE patterns. As shown, the vast majority of patterns (1057 in bin 0) are *rigid*. Only 18% of PROSITE patterns contain one or more *variable length gaps*. For the purposes of this investigation, Splash was used in a deterministic mode which identifies only rigid patterns. Heuristic extensions for the identification of flexible patterns are available.

Many PROSITE patterns are *promiscuous*, i.e., contain *tokens* that are supersets of the *substitution sets* we define in Table 1. We can “project” a PROSITE pattern onto a *substitution pattern* by replacing all *promiscuous tokens* by a *wildcard*. The idea is to assess how many patterns in PROSITE would still be *dense* enough that they could be discovered by Splash, even after having been projected onto a substitution pattern. In that case, the deterministic nature of the algorithm guarantees their discovery. We studied projected PROSITE patterns with respect to the *density constraint* to validate our choice of density constraints for the motif discovery procedure.

In Figure 3, we histogram the number of projected PROSITE patterns that satisfy the density constraint, as a function of the pattern length l , for a number of tokens $k = 2, 3, 4, 5$. The cumulative (curve) is also plotted. The first bin, not included in the cumulative, includes projected patterns that contain fewer than k tokens. If a pattern is flexible, only the most dense rigid subcomponent is analyzed. In Figure 3(a), we start by analyzing the 989 projected patterns that have at least $k = 5$ tokens. Of these, 974 (98.5%) satisfy the density constraint ($k = 5, l = 25$). Of the 320 patterns which either contain fewer than 5 tokens or are too sparse, we analyze the 151 that have 4 tokens in Figure 3(b). Of these, 128 (85%) satisfy the density constraint ($k = 4, l = 20$); 192 patterns remain undiscoverable with this density constraint. In

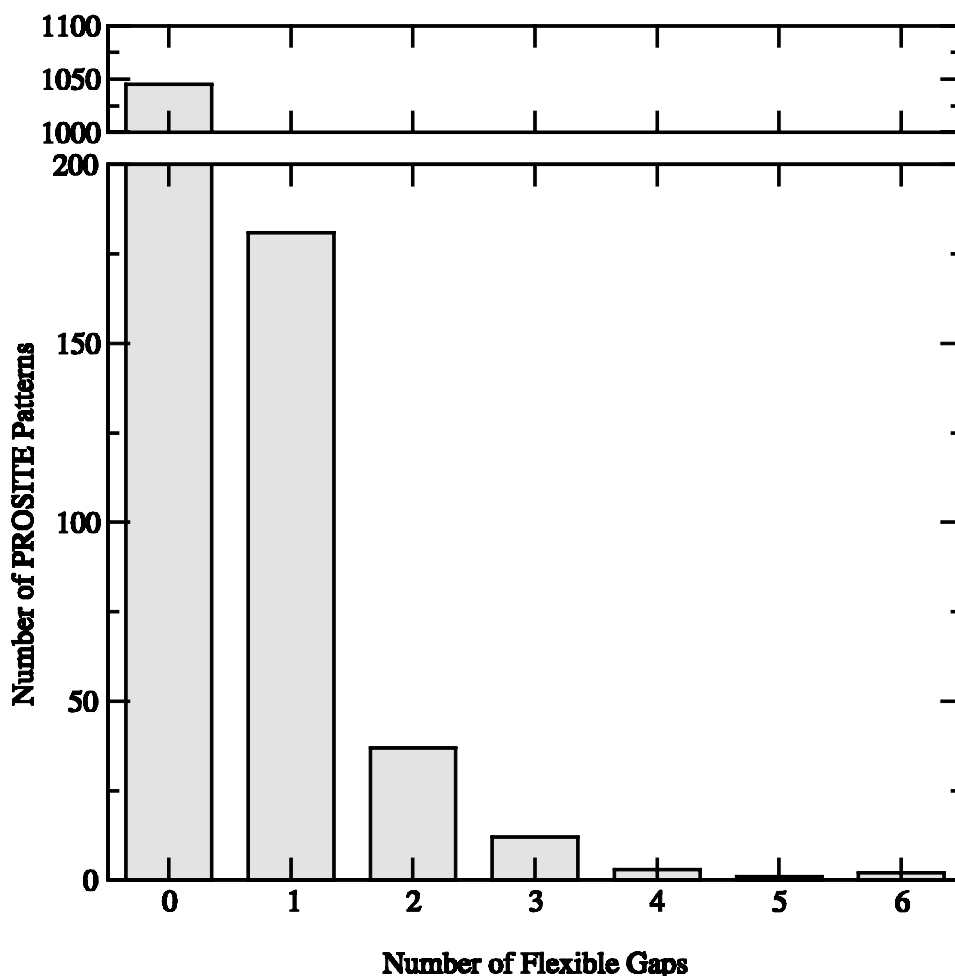


FIG. 2. Histogram of the number of variable length gaps in 1281 PROSITE patterns. Rigid patterns correspond to zero variable length gaps and constitute 82% of all PROSITE patterns.

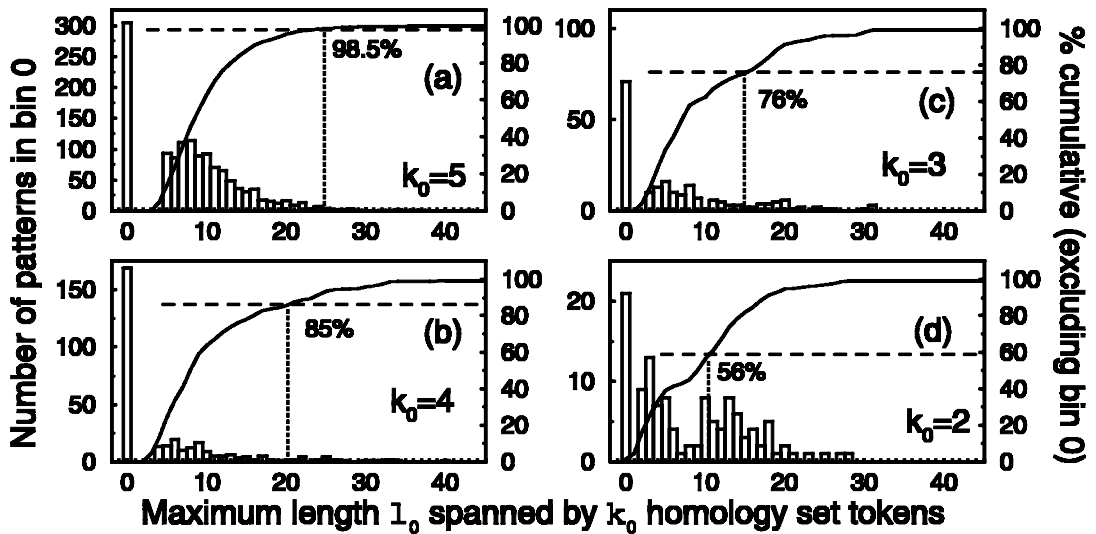


FIG. 3. Number (histograms) and cumulative (curves) of patterns that satisfy a $\langle k, l \rangle$ density constraint plotted as a function of l . Dashed lines show the cumulative at the reported threshold. In (a)–(d), bin 0 is the number of patterns which fail to meet the specified density constraint; (b)–(d) show the density statistics from bin 0 in (a)–(c), respectively. (a) 320 PROSITE patterns depicted in bin 0 fail the density constraint ($k = 5, l = 25$) and would not be discoverable with this stringency. (b) The patterns in bin 0 of (a) are screened for the constraint ($k = 4, l = 20$); 151 remain. (c) The 151 patterns in bin 0 of (b) are evaluated for the constraint ($k = 3, l = 15$). (d) 121 patterns in bin 0 of (c) are evaluated for the constraint ($k = 2, l = 10$). In (d), 64 PROSITE patterns remain undiscoverable with the constraint ($k = 2, l = 10$).

Figure 3(c), we analyze 121 of these that have at least 3 tokens. Of these, 92 (76%) satisfy the density constraint ($k = 3, l = 15$). There are only 100 projected PROSITE patterns that remain undiscoverable with the stringency ($k = 3, l = 15$). In Figure 3(d), we analyze this last set. Of these 56 (56%) satisfy the density constraint ($k = 2, l = 10$). Of the remaining 44 patterns, the 21 in the first bin are undetectable by our automatic pattern discovery algorithm because they contain either one or no solid tokens in each one of their rigid components. In all cases, we have used a window size $l = 5k$. This suggests that by performing pattern discovery with an appropriate choice of density constraint, one could find 925/989 (93%) of projected PROSITE patterns.

3.2. Sensitivity and specificity

We report the results of automated pattern discovery applied to all 974 families in PROSITE 15.0 which are associated with one or more PROSITE patterns that have a “Data bank Reference” (DR) field. For each family, the procedure described in Section 2.4 was performed with the following choice of parameters: $k = 4, l = 8, 16, 32$, and support $j = 1.0N, 0.95N, 0.90N, \dots, 2$, where N is the number of sequences in the target set. This choice of density is consistent with the results of Section 3.1 and could, in principle, allow the discovery of all but 169 patterns that have fewer than $k = 4$ tokens. The minimum Z-score was $z_0 = 1E + 3$. Patterns for which the expected number of matches in a random database with the same composition as SWISS-PROT 36 (74,019 sequences) exceeded 10% of the number of sequences in SWISS-PROT 36 were not further considered. The minimum number of reported patterns at which the procedure was halted was $n_0 = 100$.

For each pattern in PROSITE, we select all SWISS-PROT entries listed as true positives and false negatives in the DR field of the pattern record as the training set. Partial sequences were excluded. For PS00334, for instance, this results in a group of 31 sequences which includes 30 true positives and 1 false negative. If multiple PROSITE patterns are reported for the same PDOC family, only the pattern with the greatest overlap with the Splash pattern was used for comparison. For example, PS00639 and PS00640 belong to the same PDOC family, but we compare our results only for PS00639. For sake of clarity, we will limit ourselves only to the comparison of the single, most-conserved pattern across the entire set, both for PROSITE and for Splash. As seen in Section 3.5, the pattern discovery procedure can be used to extract more than one conserved pattern.

The first set of results compares the relative specificity (false positives) and sensitivity (false negatives) of corresponding PROSITE pattern and Splash patterns. For each Splash pattern in the *discovery set*, we compute the relative sensitivity Δn_{fn} and relative specificity Δn_{fp} with respect to the corresponding PROSITE pattern. Negative, zero, and positive values correspond to increased, equal, and decreased sensitivity or specificity of a Splash pattern with respect to a PROSITE pattern. Then, we determine if there is any Splash pattern in the discovery set for which, simultaneously, $\Delta n_{fp} \leq 0$ and $\Delta n_{fn} \leq 0$. If more than one is found, the one with the lowest n_{fn}^S is selected. If none is found, the one with the best S_p ($\alpha = 0.5$, Equation 2.5) is selected. The results are not strongly influenced by the choice of α . This process uniquely selects a single Splash pattern as the top-scoring one.

Classifications of the 974 PROSITE PDOC families based on negative, zero, or positive values of Δn_{fn} and Δn_{fp} are shown in Table 2 and Figure 4. The histogram of the scatter plot for both Δn_{fn} and Δn_{fp} is also reported. This shows that most patterns are accumulated in a few bins around the center of the plot ($\Delta n_{fn} = 0$ and $\Delta n_{fp} = 0$). The associated table shows that, for 76.3% of the families, Splash patterns perform at least as well as the corresponding PROSITE patterns (a and b superscripts in Table 2). For 28% of the families, Splash patterns strictly outperform the PROSITE patterns. By “strictly,” we mean improved sensitivity without detriment to specificity, or vice versa. If the ranking is done using S_p , the number of Splash patterns that perform as well or better than PROSITE patterns increases, as some of the 9% mixed cases (top left and lower right of Table 2) can now be compared in an objective fashion.

In order to assess the suitability of Z-score for evaluating patterns, we compute the intrafamily Z-score rank for the pattern with the best specificity in each of the families. Figure 5 shows a histogram of the Z-score rank and the cumulative percentage of top-scoring patterns. More than 90% of the top ranking patterns have either the best or the second best Z-score, and virtually all best-scoring patterns are in the top 10 by Z-score rank. This validates the use of the Z-score as a criteria for automatic pattern selection.

3.3. Successes and failures

Of the 273 families where Splash patterns have better or equivalent sensitivity and specificity compared to PROSITE patterns (underlined in Table 2), 178 have an overlap of at least 70% between the two patterns. These are clear instances where our procedure suggests refinement of the biolog-

TABLE 2. SUMMARY OF RELATIVE SENSITIVITY AND SPECIFICITY OF PROSITE AND SPLASH PATTERNS¹

Relative specificity (Δn_{FP})	Relative sensitivity (Δn_{FN})			Total %
	<0 %	=0 %	>0 %	
>0	22	85	10	117
%	2.3	8.7	1.0	12.0
=0	<u>171</u> ^b	470 ^a	49	690
%	<u>17.6</u>	48.3	5.0	70.8
<0	<u>50</u> ^b	<u>52</u> ^b	65	167
%	<u>5.1</u>	<u>5.3</u>	6.7	17.1
Total	243	607	124	974
%	24.9	62.3	12.7	100

¹The labels ‘<0’, ‘=0’, and ‘>0’ indicate, respectively, superior, equivalent, or worse sensitivity or specificity of Splash patterns relative to PROSITE patterns.

^aThe cell ($\Delta n_{fn} = 0$ and $\Delta n_{fp} = 0$) denotes families in which Splash and PROSITE achieve equivalent sensitivity and specificity.

^bThe underlined numbers denote cases in which Splash patterns are strictly better than PROSITE patterns. A scatterplot of all Δn_{fp} and Δn_{fn} appears in Figure 4.

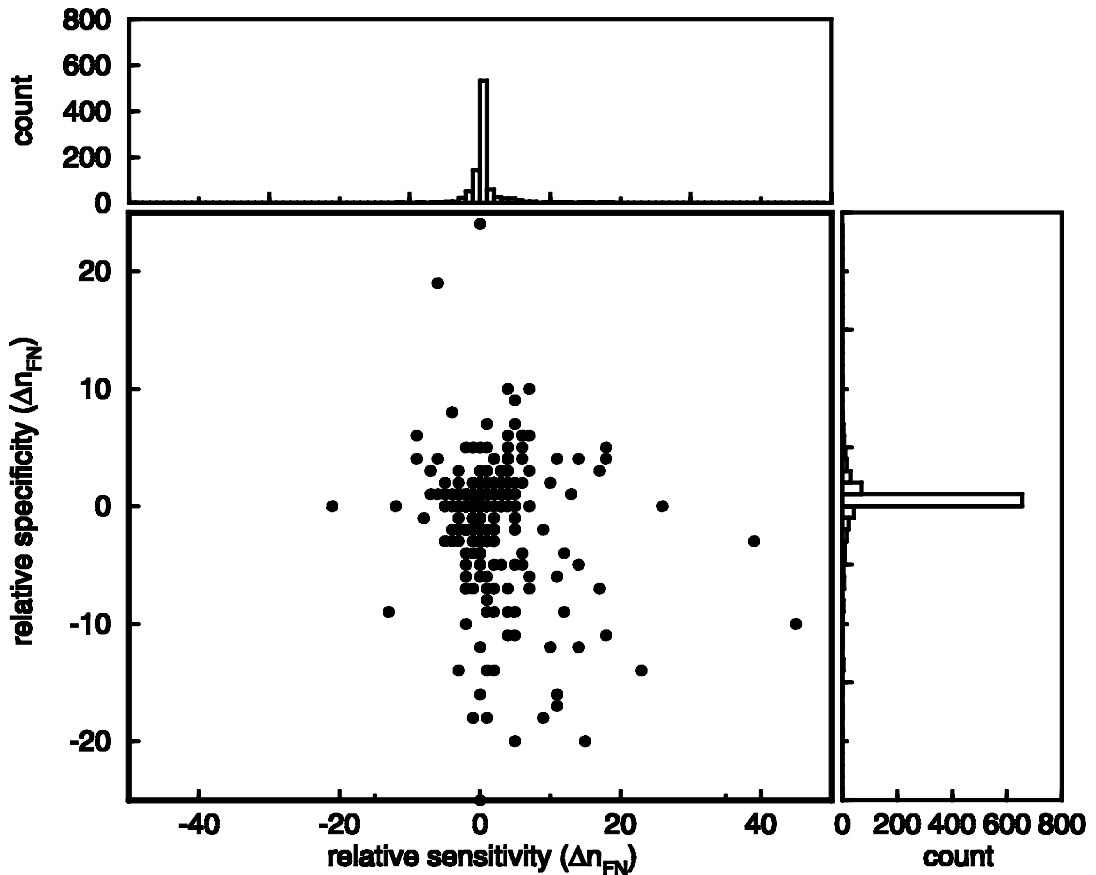


FIG. 4. Scatter plot of Δn_{fn} and Δn_{fp} for 945 of the 974 PDOC families studied; 29 families lie outside the ranges shown. A summary of the data is in Table 2. The lower left quadrant of the scatter plot (and underlined data of Table 2) represents the 76.3% of the patterns in which Splash patterns “meet or beat” PROSITE patterns in sensitivity, specificity, or both. These families are amenable to curation by the automated pattern discovery process described herein.

ically relevant PROSITE patterns that improve their sensitivity and/or specificity. Table 3 lists a few examples.

Analysis of protein families where Splash patterns exhibit worse relative specificity or sensitivity than the PROSITE patterns reveals two primary causes: flexibility and promiscuity. There are 226 (23% of 974; see Section 3.1) PDOC family records with flexible patterns in the PROSITE database. In 70 of these, Splash patterns do not score as well as PROSITE patterns. However, we note that in 78 families, Splash patterns, in spite of being rigid, have fewer or an equivalent number of false negatives and false positives. The overlap of these with the PROSITE patterns is varied.

Another significant factor in pattern performance is the extent to which promiscuity of PROSITE tokens are necessary for matching. The PROSITE database contains 982 promiscuous patterns *vis-à-vis* the substitution matrix and threshold used in this investigation. Splash permits only those occurrences in a sequence of substitution sets which achieve the support threshold; if a family exhibits diversity in residue composition which is broader than the substitution tokens in Table 1, then it will be difficult to achieve a pattern which is competitive with those in PROSITE for that family. For example, the PROSITE entry for PS00061 contains the pattern `[LIVSPADNK] .{12}Y [PSTAGNCV] [STAGNQCIVM] [STAGC] K (^PC) [SAGFR] [LIVMSTAGD] .{2} [LIVMFYW] .{3} [LIVMFYWGAPTHQ] [GSACQRH M]`, while the Splash pattern for this family is `G. [ILMV] .{4} [AS] .{12}Y. . [ANST] K`. The substitution tokens used by Splash fail to capture the extent of allowed ambiguity in some residues. Construction of a PSSM or profile HMM seeded by the Splash pattern would provide the necessary sensitivity in such instances. Of the families in which Splash patterns exhibited improved, equal, or diminished sensitivity over PROSITE patterns, 79%, 70%, and 97% respectively were families with promiscuous patterns. For specificity, these values were 87%, 69%, and 98% respectively. Thus, cases in which Splash patterns are less sensitive or specific than PROSITE

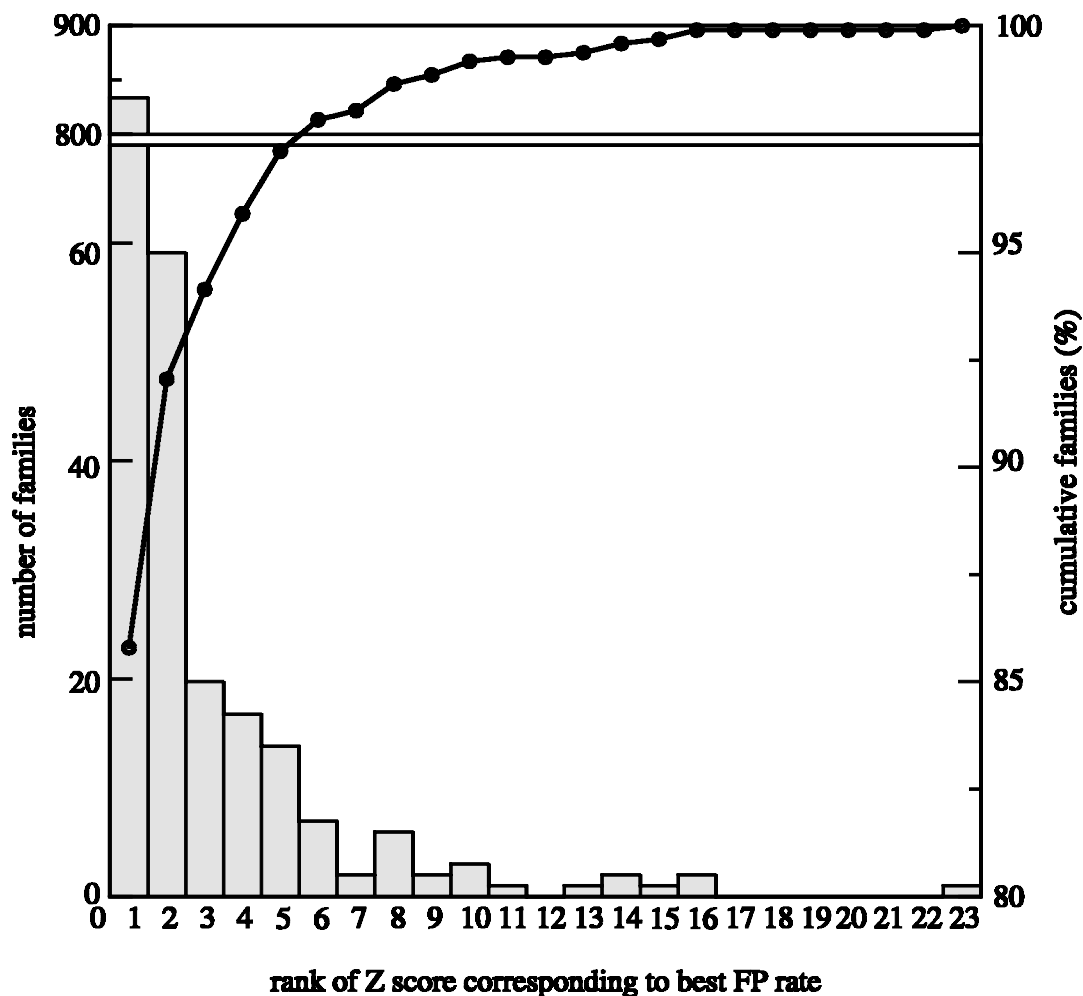


FIG. 5. Histogram of the number of PDOC families whose most specific Splash pattern corresponds to the Z-score rank shown in the abscissa. The curve indicates the cumulative percentage. For example, the most specific pattern occurs in the top 10 Z-scoring patterns for 99% of the families.

patterns are partially caused by sequence diversity greater than that expected by the substitution matrix and threshold used.

3.4. Biological significance

The following results provide a quantitative basis for the likelihood that statistically significant patterns are also biologically significant. We will assume that the patterns contained in the PROSITE database are associated with sequence regions that are important from a biological perspective. The analysis is aimed at characterizing the degree of overlap between the top-ranking Splash patterns, which as shown in the previous section have a high statistical significance, and the corresponding PROSITE pattern, which are known to have high biological significance.

Figure 6 plots how many (in percent) of the top scoring patterns (on the x axis) have an overlap larger than a given percent (on the y axis). About 72% of the Splash patterns, for families with 20 or more sequences, overlap at least 50% with their corresponding PROSITE patterns. That is, they tend to identify the same region of the protein sequence. This ratio increases to about 78% for families with at least 60 sequences. The relatively small improvement hints that, on average, fewer than about 20 sequences may be sufficient to identify biologically relevant regions. Taken together, Figures 5 and 6 demonstrate a meaningful relationship between patterns that are identified based on purely statistical criteria (Z-score) and those in PROSITE. This suggests that patterns generated by our methodology would be useful as seeds for further refinement with PSSM or profile HMM to identify biologically significant regions in protein sequences.

TABLE 3. REPRESENTATIVE EXAMPLES OF SPLASH PATTERNS THAT OVERLAP AND OUTPERFORM PROSITE PATTERNS¹

PS AC	Size	o_{ps}	n_{fn}^P	Δn_{fn}	n_{fp}^P	Δn_{fp}	Splash pattern	Family
PS00080	39	1.0	12	-12	0	0	H.H[ILMV]..[QH]...G[ILMV]	Multicopper oxidases
PS00306	28	1.0	5	-5	3	-3	M[RK][LFV][ILFV][IV][LF].C [LF].[AT]..[ILFV]A	Caesins alpha/beta
PS00490	28	1.0	10	-7	0	0	[ILMFV][ILMV]..[DE]... [ANST].[ANST]...[AS]D..L . {6} [DE]	Prokaryotic molybdopterin oxidoreductases
PS00675	64	1.0	13	-11	80	-79	[ILMV].[ILMFV].G..G.G[RQK] ...[AS]...H	Sigma-54 interaction
PS00678	259	0.9	49	-12	92	-62	[ILMFV].[ST]...D..[ILMV] [RQK].W	WD-40 repeats
PS00153	44	0.8	8	-8	1	-1	[ILMV][LFY]...[DQE].{7} [AS]...[AS]M.[ANST][AS]... N...[ILMV]...[LY]...N...Q. .[IV]T.E[IL].[DE][IV]..G	ATP synthase gamma
PS00432	205	0.8	12	-6	0	0	S...[ANST].L.[ST][LF]..[MV]. [IV].[RK].[DE][FY]	Actins
PS00027	613	0.6	37	-20	6	-1	[ILMV]...[ILMFV]..W[FWY]. N.R	Homeobox domain

¹PSAC is the PROSITE accession tag; size is the number of sequences in the family; o_{ps} , n_{fn}^P , Δn_{fn} , n_{fp}^P , and Δn_{fp} are defined in Methods. PS AC references to sequence patterns are available at <http://www.expasy.ch/prosite/>.

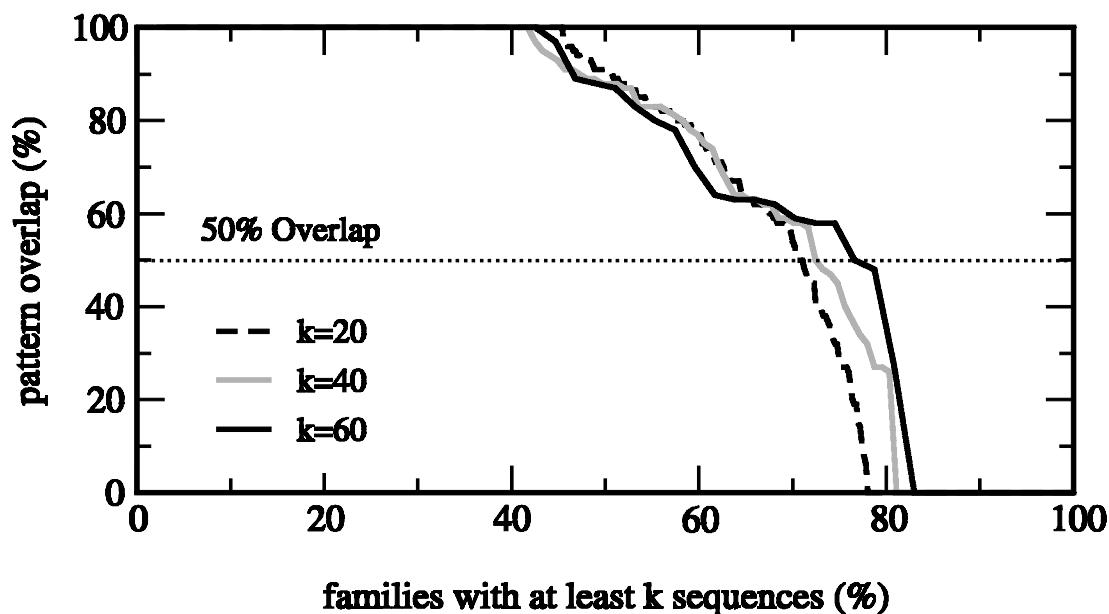


FIG. 6. The extent of overlap between PROSITE and Splash patterns for families with more than 20 (broken line), 40 (gray line), and 60 (black line) sequences. For example, 78% of the PDOC families (abscissa) with at least 60 sequences exhibit at least 50% overlap (ordinate) between PROSITE and Splash patterns.

3.5. Exhaustive pattern discovery

In the previous sections, we have purposefully limited our analysis to the single best pattern according to the S_p penalty score. However, the efficiency of the algorithm allows discovery of several patterns that are independently conserved within a family, down to very low support.

We illustrate exhaustive pattern discovery by studying the trypsin family of serine proteases. The catalytic activity of these proteases is provided by a charge relay system involving an aspartic acid residue hydrogen bonded to a histidine, which itself is hydrogen bonded to a serine, these together known as the catalytic triad. The residues forming the catalytic triad occur in well-separated regions of the sequence but are in close spatial proximity in the structure, as shown in Figure 7. PROSITE reports only two patterns for this family: PS00134 at histidine 57 and PS00135 at serine 195; there is no pattern for aspartic acid 102.

The procedure described in Section 2.6 applied to the trypsin family of 269 proteins discovers 11 patterns, each with at least 40% support. These patterns are shown in Table 4. This analysis required less than two minutes of CPU time on a 266 MHz Pentium II computer. The three patterns with the highest Z-score (1, 2, and 5) correspond to the three catalytic residues. BLOCKS reports three of these 11 patterns. Pattern 5, which contains the third conserved catalytic residue in the active site and also is one of the three highest

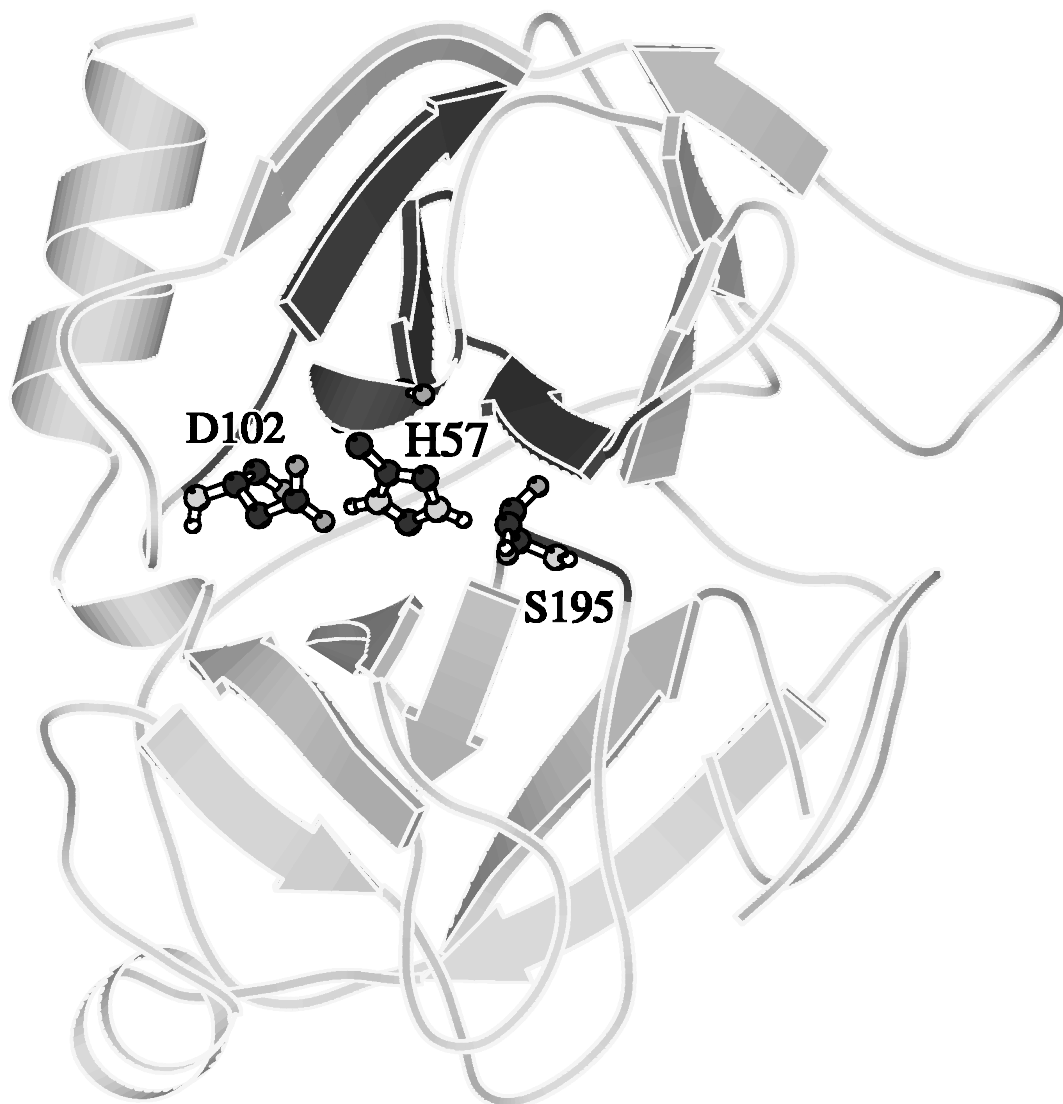


FIG. 7. Representative active site of the trypsin subfamily of serine proteases (porcine trypsin, PDB code 1aks). The aspartic acid, histidine, and serine residues of the catalytic triad are shown in ball-and-stick representation. The regions which are matched by Splash patterns for the catalytic residues (cf., Table 4) are darkened. This figure was generated with MOLSCRIPT 2.0 (Kraulis, 1991).

TABLE 4. PATTERNS FROM EXHAUSTIVE DISCOVERY OF THE TRYPSIN SUBFAMILY OF SERINE PROTEASES¹

<i>Id</i>	<i>Pattern</i>	<i>Support</i>	<i>Z-score</i>	<i>BLOCKS</i>
1	C.....[ILMV][ILMV][ST]A. (H)	268	>10.0E+300	BI0134A
2	G[DE] (S) G	274	5.93E+100	BI0134B
3	[ANST]G[HFVY]G	261	6.55E+020	
4	G.....P[HFVY]...[ILMFV]	253	5.23E+017	
5	(D) [ILMFV].L[ILMV][RQEHK][ILMV]	236	>10.0E+300	
6	G[ILMFV].[ANST].G	253	3.16E+017	
7	P..[FY]..[ILMV]....W[ILMV]	216	8.05E+057	BI0134C
8	L[RQEHK]....[ILMFV]....C	212	3.21E+006	
9	[ANST]..[ILMFV]...LP	201	4.63E+004	
10	[ILMFV]C[ANST]G	187	3.07E+003	
11	[ILMFV].L.G.[NQHY][NDHS]	139	1.27E+021	

¹Patterns 1, 2, and 5 are the highest-scoring patterns and correspond to the catalytic histidine, serine, and aspartic acid residues of serine proteases. The aspartic acid residue (pattern 5) is not identified by PROSITE or BLOCKS. Z-scores in excess of 10.0E+300 are reported as '>10.0E+300'.

Z-scoring patterns, is not reported by either BLOCKS or PROSITE. Our method of exhaustive pattern discovery and analysis of conserved regions in protein sequences, efficiently and with a rigorous statistical basis, provides a more comprehensive set of sequence motifs. This would allow the number of statistically and biologically significant motifs and profiles to be increased without a significant computational load. In particular, the deterministic nature of the pattern discovery component could further improve the sensitivity of the profiles. The collection of statistically relevant patterns could be used in conjunction to perform sequence classification.

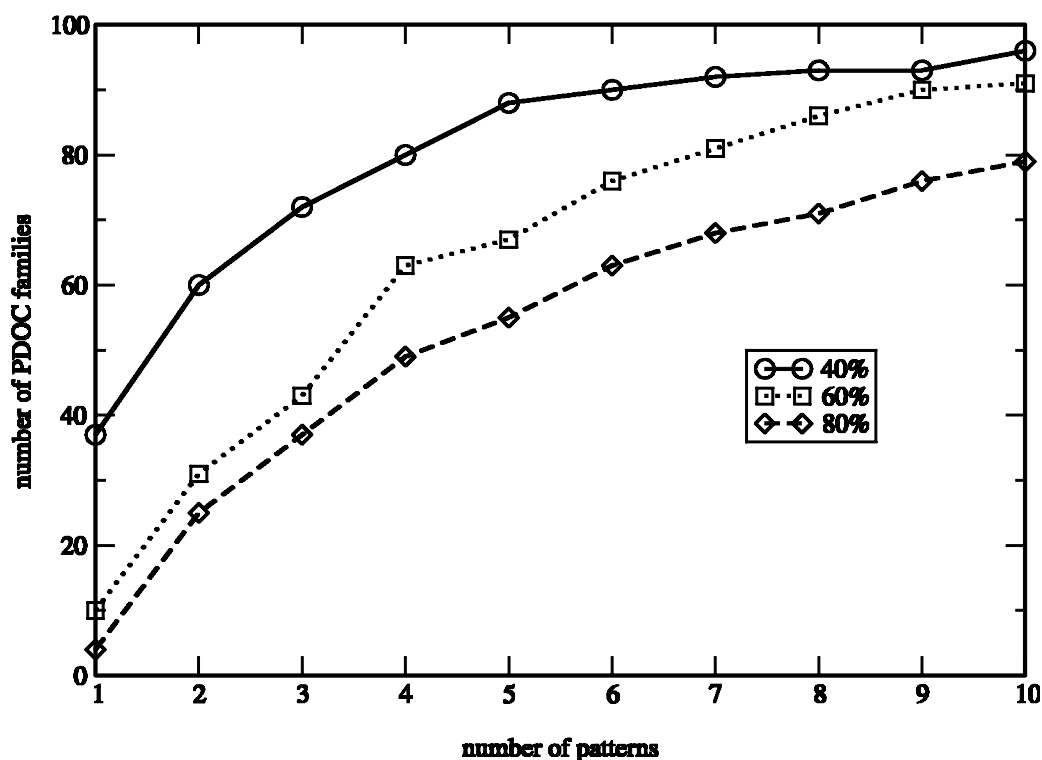


FIG. 8. Incidence of Splash patterns on regions spanned by PROSITE patterns. Data points are the number of PDOC families (ordinate) for which the top Z-scoring Splash patterns (abscissa) cover at least 40%, 60%, or 80% of the regions spanned by the PROSITE patterns.

Exhaustive pattern discovery was applied to the subset of PDOC families defined in Section 2.6. Incidence of the top Z-scoring patterns on the regions spanned by the PROSITE patterns in each PDOC family was quantified as illustrated in Figure 8. It is evident from Figure 8 that the Splash patterns with highest Z-score are typically sufficient to identify the biologically relevant regions described in PROSITE. For example, 10 patterns are sufficient to achieve 80% coverage of PROSITE patterns in 78% of the families analyzed.

4. CONCLUSIONS

We have presented a statistical framework for pattern scoring and a thorough application of the Splash algorithm for automated pattern discovery to protein sequence families in the PROSITE database. When used together, these methods provide an extremely efficient and entirely automated procedure for identifying conserved regions in a sequence family. We present a protocol of deterministic pattern discovery and demonstrate its application to automatically and successfully provide patterns that perform as well or better than those in PROSITE, for 76% of protein families in PROSITE. Splash generates patterns with better specificity and undiminished sensitivity, or vice versa, in 28% of the families; identical statistics were obtained in 48% of the families, worse statistics in 15%, and mixed behavior in the remaining 9%. A significant advantage of this approach is its efficiency and scalability. The full set of 974 protein families in PROSITE can be processed in about 12 hours of CPU time on a commodity-class workstation. This approach could significantly reduce the labor-intensive component of generating and maintaining motif databases such as PROSITE.

The Z-score statistics we have presented provide a novel method to assess the likelihood of a particular pattern. The Z-score is an assessment of the probability that a pattern of a particular composition be discovered in a random database. This is to be contrasted with a more obvious scoring system which estimates the expected number of matches as the joint probability of individual tokens. The distinction between these objectives is subtle: the Z-score incorporates the likelihood of a particular density of tokens in a pattern, in addition to the amino acid frequencies (Stolovitzky *et al.*, 1999).

By applying the method presented herein to the trypsin family of serine proteases, we obtain a set of all statistically relevant sequence motifs for a single family. Among the patterns discovered is one that contains the functionally critical aspartic acid residue and does not have a corresponding pattern in PROSITE or BLOCKS. This method may be a valuable tool to facilitate the maintenance of motif databases and significantly increase the number of biologically significant motifs (patterns or derived scoring models).

While it is conceded that scoring methods are generally more sensitive than regular expression motifs, we emphasize that identifying instances of sequence motifs does not require multiple sequence alignments and therefore may be more readily obtained. We have shown that the top Z-scoring patterns efficiently identify biologically significant regions. These patterns may be used to seed the training of scoring methods to generate very sensitive scoring models. We have used this technique to train HMMs for the *unsupervised* hierarchical classification of G-protein coupled receptors families and subfamilies with excellent results (unpublished). An elaboration of this scheme for fully automated and unsupervised family identification is being investigated.

Sequence patterns have applications in sequence alignments. A single pattern may be used to “seed” an initial alignment which is subsequently extended as in PSI-BLAST (Altschul *et al.*, 1997). Although the algorithm used to generate seed patterns in PSI-BLAST and Splash are very different, both may be used to anchor a region from which an alignment may be extended. The use of Splash patterns for sequence alignment is an open area of research which we may pursue in the future. Alternatively, a *collection* of motifs incident on a family can be assembled as a multiple sequence alignment using MUSCA (Parida *et al.*, 1999), and these alignments may be used for building scoring models.

An important use of Splash is the identification of uncharacterized conserved regions in protein sequences. In some PROSITE families, we identified patterns which have better sensitivity *and* specificity than, but do not overlap, the PROSITE pattern. These were verified to not be instances of multiple motifs within a single PDOC family. All of the results presented herein are available online at <http://www.research.ibm.com/spat/>. We are in the process of inferring patterns in collections of sequences in fold space, such as those provided by SCOP (Murzin *et al.*, 1995) and FSSP (Holm *et al.*, 1996).

ACKNOWLEDGMENTS

R.K.H. thanks the free software community for their generous contributions to the development of open standards and software. In particular, perl (www.perl.com), linux (www.linux.org), and various GNU utilities (www.gnu.org) were invaluable for this work.

REFERENCES

- Altschul, S.F., Madden, T.L., Schaffer, A.A., Zhang, J., Zhang, Z., Miller, W., and Lipman, D.J. 1997. Gapped BLAST and PSI-BLAST: A new generation of protein database search programs. *Nuc. Acids Res.* 25(17), 3389–402.
- Attwood, T.K., Flower, D.R., Lewis, A.P., Mabey, J.E., Morgan, S.R., Scordis, P., Selley, J.N., and Wright, W. 1999. PRINTS prepares for the new millennium. *Nuc. Acids Res.* 27, 220–5.
- Bailey, T.L., and Elkan, C. 1994. Fitting a mixture model by expectation maximization to discover motifs in biopolymers, in *Proceedings of the Second International Conference on Intelligent Systems for Molecular Biology*, 28–36, American Association for Artificial Intelligence.
- Bairoch, A. 1991. PROSITE: A dictionary of sites and patterns in proteins. *Nuc. Acids Res.* 19, 2241–5.
- Bateman, A., Birney, E., Durbin, R., Eddy, S.R., Finn, R.D., and Sonnhammer, E.L.L. 1999. Pfam 3.1: 1313 multiple alignments and profile HMMs match the majority of proteins. *Nuc. Acids Res.* 27, 260–2.
- Bork, P., and Koonin, E.V. 1996. Protein sequence motifs. *Curr. Op. Struct. Biol.* 6, 366–76.
- Brazma, A. 1998. Approaches to the automatic discovery of patterns in biosequences. *J. Comp. Biol.* 5, 279–305.
- Califano, A. 2000. SPLASH: Structural pattern localization analysis by sequential histograms. *Bioinformatics* 16, 341–57.
- Durbin, R., Eddy, S., Krough, A., and Mitchison, G. 1998. *Biological Sequence Analysis: Probabilistic Models of Protein and Nucleic Acid*. Cambridge University Press.
- Gribskov, M., McLachlan A.D., and Eisenberg, D. 1987. *Proc. Natl. Acad. Sci. USA* 84, 4355–8.
- Henikoff, S., and Henikoff, J.G. 1991. Automated assembly of protein blocks for database searching. *Nuc. Acids Res.* 19, 6565–72.
- Henikoff, S., and Henikoff, J.G. 1992. Amino acid substitution matrices from protein blocks. *Proc. Natl. Acad. Sci. USA* 89, 10915–9.
- Henikoff, S., Henikoff, J.G., and Pietrokovski, S. 1999. Blocks+: A nonredundant database of protein alignment blocks derived from multiple compilations. *Bioinformatics* 15, 471–9.
- Holm, L., and Sander, C. 1996. Mapping the protein universe. *Science* 273, 595–602.
- Jonassen, I. 1997. Efficient discovery of conserved patterns using a pattern graph. *Comp. Appl. Biosci.* 13, 509–22.
- Kraulis, P.J. 1991. MOLSCRIPT: A program to produce both detailed and schematic plots of protein structures. *J. Appl. Cryst.* 24, 946–950.
- Murzin, A.G., Brenner, S.E., Hubbard, T., and Chothia C. 1995. SCOP: A structural classification of proteins database for the investigation of sequences and structures. *J. Mol. Biol.* 247, 536–40.
- Neuwald, A.F., and Green, P. 1994. Detecting patterns in protein sequences. *J. Mol. Biol.* 239(5), 698–712.
- Neuwald, A.F., Liu, J.S., and Lawrence, C. 1995. Gibbs motif sampling: Detection of bacterial outer membrane protein repeats. *Prot. Sci.* 4, 1618–32.
- Nevill-Manning, C.G., Wu, T.D., and Brutlag, D.L. 1998. Highly specific protein sequence motifs for genome analysis. *Proc. Natl. Acad. Sci. USA* 95, 5865–71.
- Parida, L., and Rigoutsos, I. 1999. An approximate algorithm for alignment of multiple sequences using motif discovery. *J. Comb. Opt.* 3, 247–75.
- Rigoutsos, I., and Floratos, A. 1998. Combinatorial pattern discovery in biological sequences: The TEIRESIAS algorithm. *Bioinformatics.* 14(1), 55–67.
- Sonnhammer, E.L.L., and Kahn, D. 1994. Modular arrangement of proteins as inferred from analysis of homology. *Protein Sci.* 3, 482–492.
- Stolovitzky, G., and Califano, A. 1999. Pattern statistics in biological datasets. *IBM Research Communication*. <http://www.research.ibm.com/kplash/Papers/Pattern%20Statistics.pdf>
- Wall, L., Christiansen, T., and Orwant, J. 2000. *Programming Perl*, 3rd edition. O'Reilly and Associates, Sebastopol, CA.

Address correspondence to:
Ajay K. Royyuru
IBM Computational Biology Center
P.O. Box 218
Yorktown Heights, NY 10598

E-mail: ajayr@us.ibm.com

This article has been cited by:

1. Tao Tao, Cheng Xiang Zhai, Xinghua Lu, Hui Fang. 2004. A Study of Statistical Methods for Function Prediction of Protein Motifs. *Applied Bioinformatics* **3**:2, 115. [[CrossRef](#)]
2. Agatha H. Liu, Andrea Califano. 2003. CASTOR: Clustering Algorithm for Sequence Taxonomical Organization and Relationships. *Journal of Computational Biology* **10**:1, 21-45. [[Abstract](#)] [[PDF](#)] [[PDF Plus](#)]
3. 2001. Current Awareness. *Comparative and Functional Genomics* **2**:3, 187. [[CrossRef](#)]