Hidden Markov Models for Detecting Remote Protein Homologies

Kevin Karplus, Christian Barrett, Richard Hughey
Department of Computer Engineering
Jack Baskin School of Engineering
University of California
Santa Cruz, CA 95064

PREPRINT to appear in Bioinformatics, 1999

Abstract

Motivation

A new hidden Markov model method (SAM-T98) for finding remote homologs of protein sequences is described and evaluated. The method begins with a single target sequence and iteratively builds a hidden Markov model (HMM) from the sequence and homologs found using the HMM for database search. SAM-T98 is also used to construct model libraries automatically from sequences in structural databases.

We evaluate the SAM-T98 method with four datasets. Three of the test sets are fold-recognition tests, where the correct answers are determined by structural similarity. The fourth uses a curated database. The method is compared against WU-BLASTP and against DOUBLE-BLAST, a two-step method similar to ISS, but using BLAST instead of FASTA.

Results

SAM-T98 had the fewest errors in all tests—dramatically so for the fold-recognition tests. At the minimum-error point on the SCOP-domains test, SAM-T98 got 880 true positives and 68 false positives, DOUBLE-BLAST got 533 true positives with 71 false positives, and WU-BLASTP got 353 true positives with 24 false positives.

The method is optimized to recognize superfamilies, and would require parameter adjustment to be used to find family or fold relationships.

One key to the performance of the HMM method is a new score-normalization technique that compares the score to the score with a reversed model rather than to a uniform null model.

Availability

A World Wide Web server, as well as information on obtaining the Sequence Alignment and Modeling (SAM) software suite, can be found at `http://www.cse.ucsc.edu/research/compbio/`

Contact

`karplus@cse.ucsc.edu`
`http://www.cse.ucsc.edu/~karplus`

1 Introduction

A critical task confronting genome sequencing projects today, and biology in general, is the functional and structural characterization of new proteins. Characterization is often inferred by similarity to proteins of known structure or function whose amino acid sequences have diverged through mutation. Finding these evolutionary connections, which can be difficult to detect in distantly related proteins, is called *remote-homolog detection*. Methods that are reliably able to detect subtler similarities between sequences are thus able to assign putative structure and functional characterization to more new proteins.

The focus of this paper is to present a new hidden Markov model (HMM) method to detect remote homologies. The SAM-T98 method creates a hidden Markov model from a single target sequence by iteratively finding homologs in a protein database and refining the model. We compare our results to those using more established methods.

The results are presented in the context of four tests, three of which are fold-recognition tests. These three tests use a set of *target* sequences whose folds are to be determined, a *fold database* of sequences of known structure, and a definition of "correct" target-database sequence pairings. The fourth uses a curated database whose protein sequences were grouped according to family, primarily using sequence information. For all of the tests, we used only primary sequence information—the test was purely one of detecting remote homologs, not of

protein structure prediction or threading.

For the fold-recognition tests, our HMM-based methods did extremely well at all levels of acceptable error, finding many more remote homologs than the more traditional sequence-based methods.

A companion paper (Park *et al.*, 1998) compares SAM-T98 on the SCOP test sets with BLAST (Altshul *et al.*, 1990) and FASTA (Pearson & Lipman, 1988) and with two state-of-the-art methods: PSI-BLAST (Altschul *et al.*, 1997) and ISS (Park *et al.*, 1997). The results there show SAM-T98 to be superior to PSI-BLAST, which is superior to ISS, which is superior to BLAST and FASTA.

1.1 Hidden Markov models

Profile hidden Markov models (Haussler *et al.*, 1993; Krogh *et al.*, 1994) or generalized profiles (Bucher & Bairoch, 1994) have been demonstrated to be very effective in detecting conserved patterns in multiple sequences (Hughey & Krogh, 1996; Baldi *et al.*, 1994; Eddy *et al.*, 1995; Eddy, 1995; Bucher *et al.*, 1996; McClure *et al.*, 1996; Karplus *et al.*, 1997; Grundy *et al.*, 1997; Karchin & Hughey, 1998). The typical profile hidden Markov model (Figure 1) is a chain of match (square), insert (diamond), and delete (circle) nodes, with all transitions between nodes and all character costs in the insert and match nodes trained to specific probabilities. The single best path through an HMM corresponds to a path from the Start state to the End state in which each character of the sequence is related to a successive match or insertion state along that path (delete states indicate that the sequence has no character corresponding to that position in the HMM).

For this work we use a local alignment procedure that relates part of the sequence to one contiguous path through part of the HMM (Tarnas & Hughey, 1998). If two sequences are aligned to the model, a multiple alignment between those sequences can be inferred from their alignments to the model, though it must be remembered that characters modeled by insert states are not aligned between sequences.

When an HMM is trained on sequences that are members of a protein family, the resulting HMM can identify the positions of amino acids which describe conserved primary structure of the family. This HMM can then be used to discriminate between family and non-family members in a search of a sequence database. A multiple alignment of sequences to the HMM will reveal the regions in the primary structure that are conserved and that are characteristic of the family.

2 Test Sets

The first three test sets use databases of sequences of known structure to provide a measure of relatedness between the structures. In each case, we had a set of target sequences whose fold we wanted to determine by matching it against all the sequences in the fold database. We evaluated how the methods described in the next section discriminated between the homologous and nonhomologous sequences in the database for all of the target sequences.

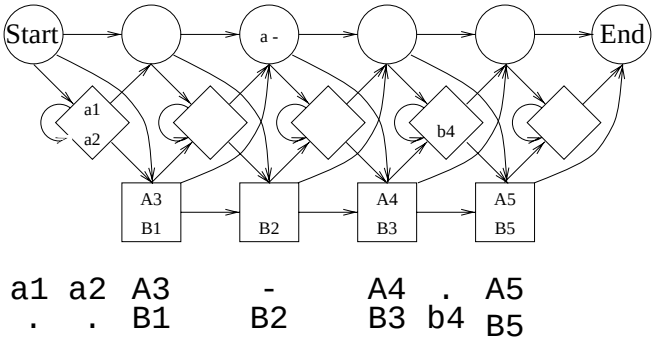


Fig. 1. An example of an HMM with two sequences whose characters are generated by the HMM, and the corresponding alignment. Positions modeled by the HMM's match states are indicated with uppercase letters, while those modeled by unaligned insertion states are indicated with lowercase letters.

2.1 FSSP

The FSSP test set is based on the July 1997 FSSP protein classification tree (Holm & Sander, 1996; Holm & Sander, 1997). Our fold database contains the sequences of all 1050 leaves of the FSSP tree, and our target list is a subset of 166 sequences chosen arbitrarily to cover all major subtrees. The use of the FSSP tree ensures that no two sequences in the database have more than 25% identical residues in the correct structural alignment.

The FSSP test set uses DALI structure comparison (Holm & Sander, 1993) to determine structural homology. A *soft-threshold* classification was made, in which DALI z-scores higher than 6 were considered to be homologs, z-scores lower than 2 were non-homologs, and z-scores between 2 and 6 were counted as partly homologous and partly non-homologous using a linear interpolation to get a homology score between 0 and 1. For the 174,134 non-self pairs, the sum of the homology scores was 3510.85 (so about 2% of the pairs represent homologies to be detected), though the best possible classifier still makes at least 1494.95 errors (Figure 2). At the minimum-error point for an optimal classifier, there are 2449.45 homolog pairs (1.4% of the possible pairs).

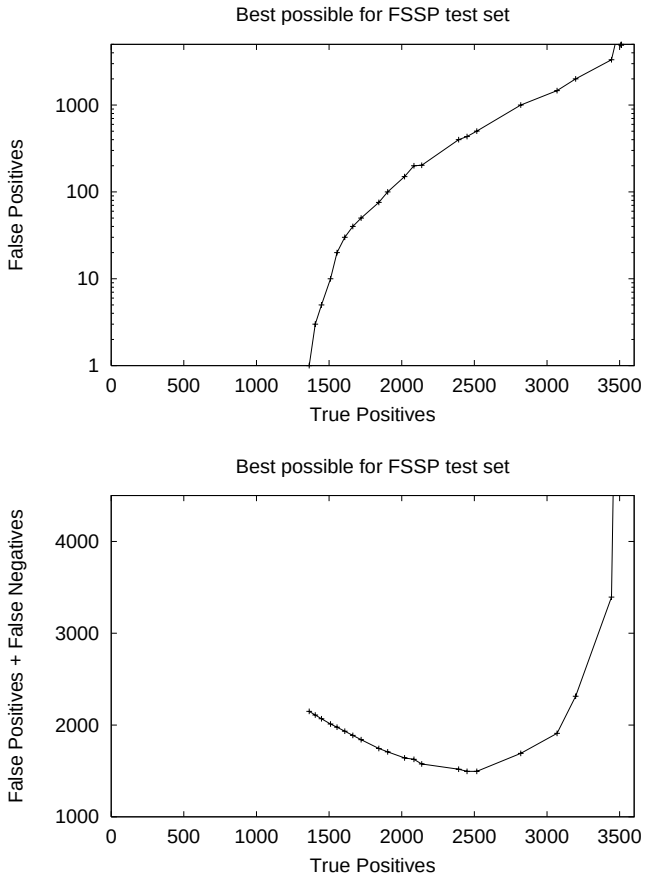


Fig. 2. The best possible number of false positives (top) and the errors as a function of the number of true positives for the soft-thresholding done in the FSSP test.

2.2 SCOP

We used two test sets (Brenner, 1996; Park *et al.*, 1997) derived from the Structural Classification of Proteins (SCOP) hierarchy (Hubbard *et al.*, 1997). For each test set, we used identical lists for both the target list and the database of known folds. A pair was labeled as homologous if both sequences were in the same SCOP superfamily, otherwise it was labeled as nonhomologous. No two sequences in either test set had more than 40% sequence similarity. In a related paper (Park *et al.*, 1998), pairs in the same fold but different superfamilies were not counted as either correct or incorrect. Doing so would reduce our number of false positives by only 3 at the minimum error point (1eit-1tabI, 1lab-1gpr, 1iva-1tabI), so we did not feel it necessary to add this ambiguity.

The *whole-chain* test set was composed of 571 single-domain proteins. Of the 162,735 pairings, only 931 (0.6%) are considered correct homologies. The *domain* test set contained the whole-chain test set, plus another 364 domains that were only parts of chains (935 sequences in total). Of the 436,645 possible non-self pairs, only 2605 were considered homologs (0.6%).

The higher rate of homology for the FSSP dataset may be an artifact of our selecting target sequences to cover the major subtrees—sequences with few true positives were less likely to be picked as targets.

2.3 Pearson

The fourth test is Pearson's test for sequence-comparison tools (Pearson, 1995). It is a curated version of the PIR database (PIR1, release39) (Barker *et al.*, 1990) augmented with 237 sequences. In total, it contains 12,216 sequences.[1] We report results here for the "e0" set of 67 target sequences chosen by Pearson. Of the 818,405 possible non-self pairs of the target sequence with database sequences, 3474 (0.4%) were considered correct.

Since the PIR families are generally of fairly close homologs, the Pearson test set is a test of close-homolog classification, not remote homolog classification.

3 Algorithms

We evaluated three remote-homology detection methods: two based on WU-BLASTP and one based on HMM methods using the SAM software. We used WU-BLASTP version `2.0a16MP-WashU` (available from `http://blast.wustl.edu`) and SAM suite version 2.0 (Hughey & Krogh, 1996), although the latter was not used with default parameters. For example, different transition regularizers were specified and noise and model surgery were not used. Additionally, local alignments, not global alignments, were used.

3.1 Blast-based methods

Two of the methods used here are based on the BLAST search program (Altshul *et al.*, 1990), perhaps the most widely used bioinformatics tool today. This program is extremely fast and easy to use, so evaluating it is essential. Tools that fail to outperform BLAST are rarely worth their computational cost.

3.1.1 WU-Blast

The simplest approach to remote-homolog detection is to provide the target sequence to a version of BLAST, and collect the top hits in the database. In order to allow us to sweep the threshold over a wide range, we set the `E` parameter (the expected number of false positives) to 10 for each search. We recorded the logarithm of the reported P-value as the score to threshold. The exact

[1] There are 12,219 sequences in the database, but three of them are duplicates of others.

setting of E is probably unimportant, as the optimum threshold never corresponded to a P-value greater than 0.005.

3.1.2 Double-Blast

The DOUBLE-BLAST method was inspired by ISS (Park *et al.*, 1997). No direct comparison with ISS is included here, but comparisons have been done on the two SCOP datasets (Park *et al.*, 1998), and DOUBLE-BLAST appears to be similar to ISS in the effectiveness of its searches.

Instead of trying to find the homologs in the database directly from the target sequence, a two-step approach is used. First, a set of close homologs to the target sequence is found in a large database of sequences, then each homolog is used as a query to search the final database. The large database employed is the non-redundant protein database NRP (NRP, 1998). WU-BLASTP is used both for finding the set of close homologs and for using each of these homologs to perform the second search. The first search is done with an E-value of 0.00005, and the second search with an E-value of 0.2. The score reported is the log of the maximum of the reported E-values for the hits. Each hit found in the first search is treated as a separate homolog, as attempts to combine the hits resulted in many more false positives. This was particularly evident for the SCOP whole-chain test set, since non-homologous domains may occur between two homologs in a database sequence.

3.2 The SAM-T98 HMM method

Presented with a single target sequence, the SAM-T98 method attempts to find and multiply align a set of homologs and then create an HMM from that multiple alignment. The resulting HMM is then used for database search. The construction, training, and application of the HMMs is all done with programs from the SAM package (Hughey & Krogh, 1996).

When the database is small, the SAM-T98 method can also be used to create an HMM for each sequence in the database. This database of models can then be searched with the target sequence, providing a two-pronged approach to the search problem. Because SAM-T98 iteratively creates a model from a single sequence, hand-tuned seed alignments, such as those used for PFAM (Sonnhammer *et al.*, 1997), are not needed, though the method could be applied to such seed alignments.

For the fold-recognition tests, we created HMMs for all of the sequences in the fold database (1050 for FSSP and 931 for SCOP, 1677 in all, taking the overlap into account). For the Pearson test, since we were unwilling to build an HMM for each of the 12,216 sequences in the database, we used SAM-T98 to build HMMs only for the 67 target sequences, and scored with just the target HMMs. Based on the results for the other test sets, using only target HMMs reduces performance only slightly (see Summing Scores, below).

Since building HMMs from weighted multiple alignments is a critical aspect of the method, we specifically discuss sequence weighting next, followed by the SAM-T98 method itself and a discussion on how the HMMs were used to score sequences in the test sets.

3.2.1 Weighting sequences

The SAM-T98 method uses sequence weighting for building models from alignments, both internally and when the final alignments are used to create the models for scoring a set of sequences.

The relative weights are set with the Henikoffs' position-based sequence weights (Henikoff & Henikoff, 1994), but the absolute weight is set to get a specific level of entropy averaged over all columns after a Dirichlet mixture regularizer (Sjölander *et al.*, 1996) is applied to the weighted counts. The entropy is specified by the number of bits saved relative to the entropy of the background distribution. This relative entropy measure has been used previously to characterize substitution matrices (Altschul, 1991), and the popular BLOSUM50 and BLOSUM62 matrices corresponds to saving about 0.5 and 0.7 bits per column. The savings for our method varies from 2.5 bits for alignments with only 20 match columns down to about 0.36 bits per column for alignments with over 600 match columns. More precisely, the savings requested for an n-column alignment is $50/\min(n, 140(1 - e^{-0.008n}))$, where n is the length of the alignment.

The large savings requested for short alignments is generally not available with any weights, and the relatively poor performance of the SAM-T98 method on short peptides, noticeable when analyzing the top false positives for the SCOP domain test set, may be due to this weighting problem.

3.2.2 The SAM-T98 method

SAM-T98 starts with a query sequence and searches the non-redundant protein database using WU-BLASTP to produce two sets of potential homologs: one of very close homologs ($E < 0.00003$) and one of possible homologs ($E < 500$). The initial WU-BLASTP cull of NRP is necessary for two reasons: we do not expect an HMM built from a single sequence to do as well at finding homologs as WU-BLASTP does, and an HMM database search of all of NRP is too slow for building thousands of alignments.

The SAM-T98 method then uses 4 iterations of a selection, training, and alignment procedure. For each iteration it needs an initial alignment, a set of sequences to search, a threshold value, and a transition regular-

izer. From the alignment and regularizer, an HMM is constructed and used to score the set of sequences. All sequences that score better than the threshold value are used to estimate a new HMM. Alignment of the training sequences to the HMM produces the alignment that is the input for the next iteration.

On the first iteration the single sequence passed to the method is used as the initial (trivial) alignment and the close homologs found by WU-BLASTP are used as the search set. The threshold is set strictly (-40 nats), so only strong matches to the sequence are considered. The transition regularizer approximates the gap costs used by WU-BLASTP. Requiring both WU-BLASTP and the initial HMM to score a sequence well ensures that only close homologs are included at this stage of the process.

On subsequent iterations the input alignment is the output from the previous iteration and the search set is the larger set of possible homologs found by WU-BLASTP. The thresholds are gradually loosened (-30 nats, -24 nats, and -16 nats).

For the second and third iteration, we use a regularizer that encourages long sequences of match states, and for the final iteration a transition regularizer trained on FSSP structural alignments is used.

The above selection, training, and alignment procedures consists of several calls to SAM programs. Models are created with SAM's `modelfromalign` program which uses the alignment, sequence weighting, transition regularizer, and Dirichlet mixture to build an HMM. Scoring the sequence set with an HMM uses SAM's multiple domain scoring procedure, now part of `hmmscore`, which selects only the portion of a sequence matching the HMM (local scoring (Smith & Waterman, 1981) as applied to SAM models (Tarnas & Hughey, 1998)). From the sequences selected using this procedure, a new model is estimated using SAM's `buildmodel` HMM training program. The alignment of the training sequences back to the resulting HMM is accomplished with SAM's `align2model` program. To ensure that the initial sequence to the whole process is not lost, it is added to the training set at this point, and any duplicate sequences in the training set are eliminated.

Since this process is involved and requires substantial computing time, it is only done once for any sequence and the final alignment is kept as an entry in a library. An HMM can be quickly constructed for the stored alignment using `modelfromalign` and sequence weighting.

The quality of the HMM resulting from this method is critically dependent on the sequences selected for training, and this sequence selection depends on the scoring implementation. During the method's development, we found that many protein families' multiple alignments show columns of strict conservation of what are usually the more rarely seen residues (cysteine, for example). When scoring databases with an HMM built for these families, sequences that are compositionally biased toward these residues tend to receive inflated scores and become false positives.

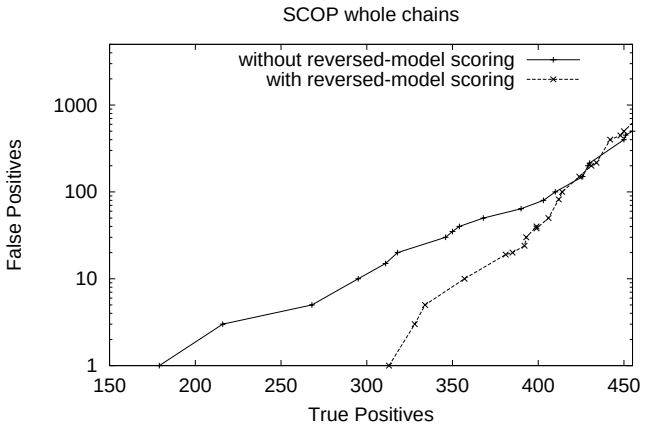


Fig. 3. Incorporating reversed-model scoring into the SAM-T98 method's iterative procedure results in HMMs that are better homolog discriminators than using a standard null model. This is illustrated here using the SCOP whole-chain test set.

Before this observation, scoring involved comparing the log-probability of a sequence for an HMM with its log-probability for a null model (Barrett *et al.*, 1997). To address this problem, we looked at the difference of the log-probability of the sequence and the log-probability of the sequence with a reversed HMM (equivalently, the score of the reversed sequence with the HMM). Since the reversed sequence has the same length and composition as the sequence, these two sources of error are effectively eliminated. Figure 3 shows the effectiveness of this reversed model scoring on the SCOP whole chain test set. For the remainder of this experiment, we used reversed model scoring when scoring an HMM against the test sets.

3.2.3 Summing scores

There are two ways to score a target against a database: one can build an HMM for the target sequence and score all database sequences using the model, or one can score the target sequence using HMMs built for all of the database sequences. We experimented using just the target model score, just the database model score, and the sum of the two scores.

To gauge the effectiveness of score summing, in Figure 4 we plot the false positives as a function of true positives using both the SCOP whole-chain and FSSP test

sets. As can be seen for the former, the added computational burden of building an HMM for all of the test set sequences so that score summing can be performed is not always justified. This changes when one considers the FSSP test set, as summing provides definite improvement beyond the 100 false positives level. This difference may be attributable to the fact that the FSSP test set contains sequences with no more than 25% sequence homology (as opposed to the SCOP whole-chain's 40%), and the summing is necessary to strengthen the weak scores between a truly homologous pair.

Another possible explanation is that the SCOP test consisted solely of single domains, while the FSSP test had to match domains from multiple-domain proteins. When the target and template have very different lengths, the scoring may well work better in one direction than the other.

For the structure-based fold-recognition tests, we performed both directions of scoring and summed the scores.

4 Results

Common to each of the test sets was a list of target (query) sequences and a database that contained homologs for each of these target sequences. A perfect homology search would cleanly separate the homologs in the database from the non-homologs. For any given threshold, we can identify the true positives (homolog pairs scoring better than the threshold), the false positives (non-homolog pairs scoring better than the threshold), and the false negatives (homolog pairs scoring worse than the threshold). An *error* is either a false positive or a false negative. The *soft-threshold* used in the FSSP test set made perfect separation impossible.

To evaluate the performance of the search methods for each test set, all pairs of target sequence and database sequence were sorted from best score to worst score. By sweeping through this sorted list, we compare the methods in three fashions. First, to make comparisons based on one number, in Table I we compare the number of errors at each method's minimum error point. Next, in Figures 5–8, discussed below, we plot the number of non-homolog pairs found versus the number of homolog pairs found (the false positives as a function of true positives). Since the number of false positives grows roughly exponentially with the number of true positives, setting an optimal threshold is difficult from the false-positive-versus-true-positive plot. Thus, we also plot the total number of errors as a function of true positives to provide a more detailed look at the tradeoff between precision (minimizing false positives) and recall (minimizing false negatives).

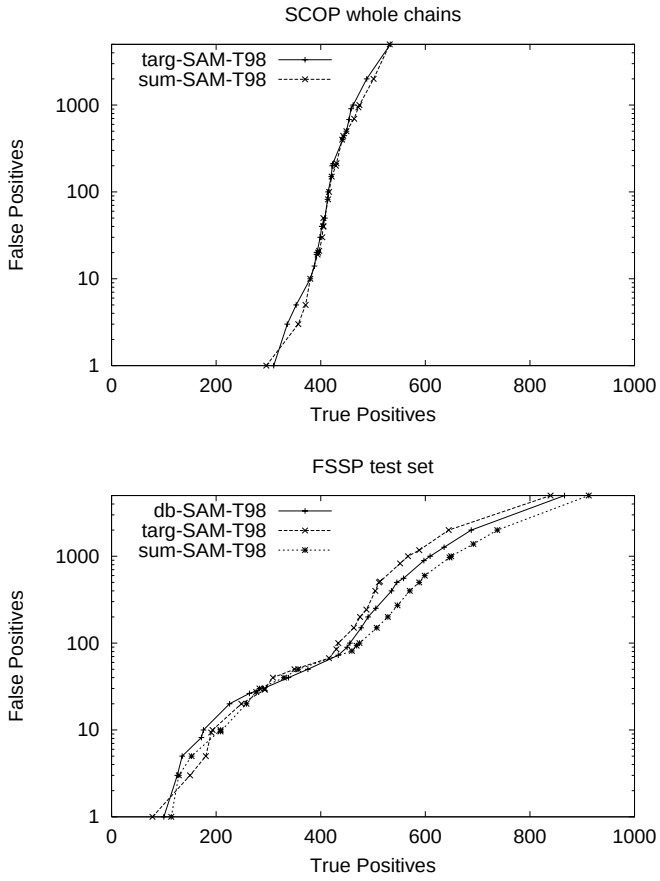
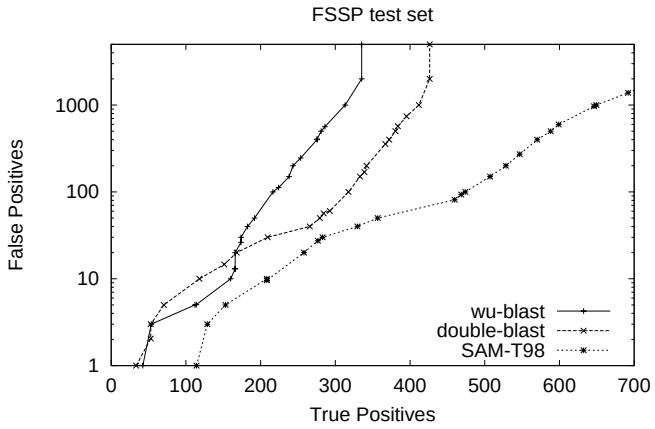
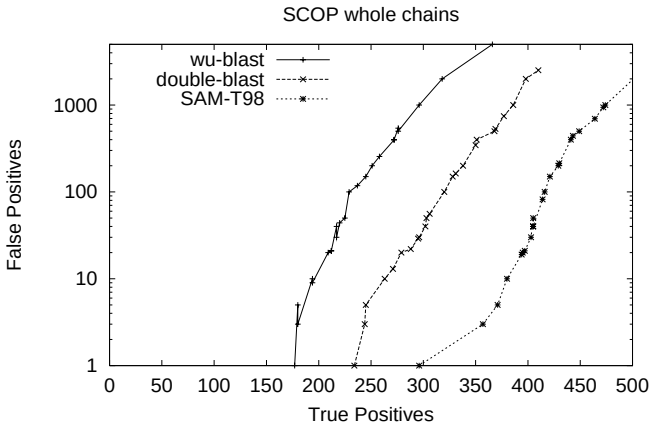


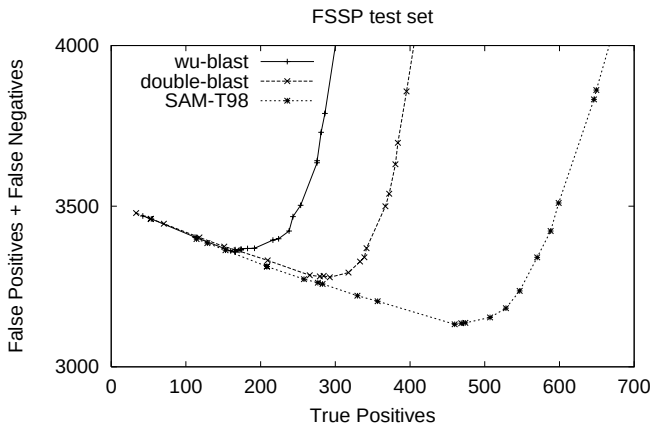
Fig. 4. Summing of scores does not provide much improvement for the SCOP whole-chain test set (top). For the FSSP test set, summing the scores (sum-SAM-T98) from the model library HMMs (db-SAM-T98) and the target sequence's HMM (targ-SAM-T98) provides an improvement beyond about 100 false positives. The symmetry of the SCOP whole-chain test makes db-SAM-T98 and targ-SAM-T98 curves identical.

4.1 FSSP

Figure 5 shows false positives and errors as functions of true positives. Both curves show the HMM-based methods doing much better than the BLAST-based methods. Because of the *soft-threshold* classification for this test set, the fewest errors any method could have achieved is 1494.95. SAM-T98 came closest to this mark with 3132.50 errors, while DOUBLE-BLAST and WU-BLASTP had 3281.55 and 3363.35, respectively.







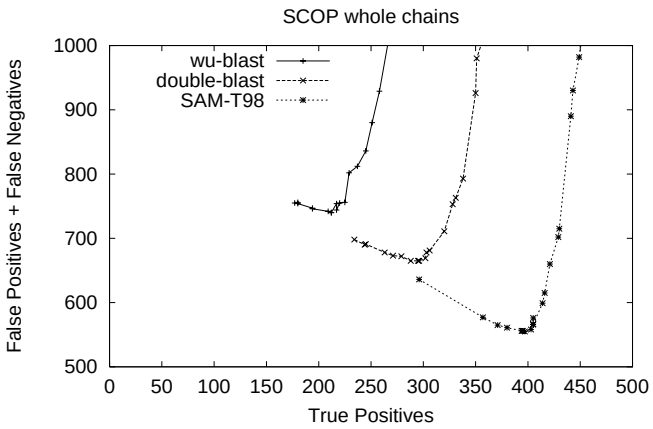


Fig. 5. Comparison of the methods for the FSSP test set. SAM-T98 distinguishes more true homologs than WU-BLASTP or DOUBLE-BLAST for any error rate. The theoretically best possible performance is shown in Figure 2.

Fig. 6. Results for the methods on the SCOP whole-chain test of 571 sequences show that SAM-T98 is a much better homolog discriminator than the other methods. The maximum possible number of true positives is 931.

4.2 SCOP

4.2.1 Whole-chain test set

For the whole-chain SCOP dataset, Figure 6 shows that the HMM-based methods perform best for all levels of false positives. If no false positives are allowed, WU-BLASTP gets 148 true positives, DOUBLE-BLAST gets 233, and SAM-T98 gets 256. The minimum-error points are even more dramatically separated with 740 for WU-BLASTP, 665 for DOUBLE-BLAST, and only 555 errors for SAM-T98 (see Table I).

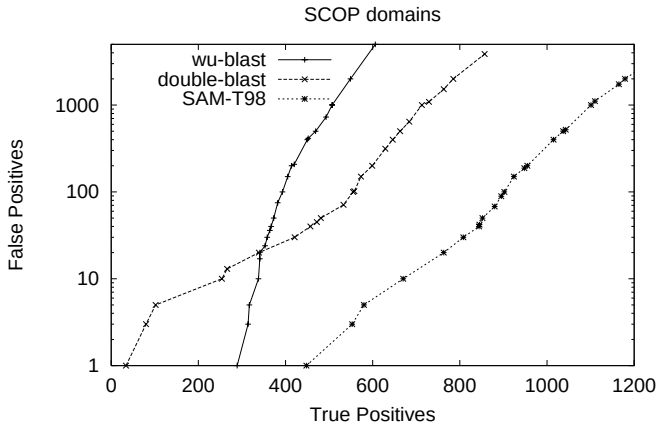
4.2.2 Domain test set

On the domain test, if the threshold is set to exclude all false positives, WU-BLASTP does best with 268 true positives, while DOUBLE-BLAST gets only 14, and SAM-T98 gets 101. The good performance of WU-BLASTP does not extend far, as SAM-T98 beats WU-BLASTP if even 1 false positive is allowed. This test set probably provides the most dramatic improvement of the HMM-based methods over the BLAST-based ones. This is particularly evident in Figure 7, where the minimum error points are 2276 for WU-BLASTP, 2143 for DOUBLE-BLAST, and 1793 for SAM-T98.

4.3 Pearson

The Pearson test set differs from the others in that the database sequences generally do not have known structure. The hand-classification of the sequences into families relies heavily on sequence similarity, resulting in families composed of generally close homologs.

The closeness of the members of the families can be seen in the excellent performance of WU-BLASTP on this dataset. With no false positives, WU-BLASTP gets 547 true positives, DOUBLE-BLAST gets 603 true positives, and SAM-T98 gets only 350. At 200 false positives



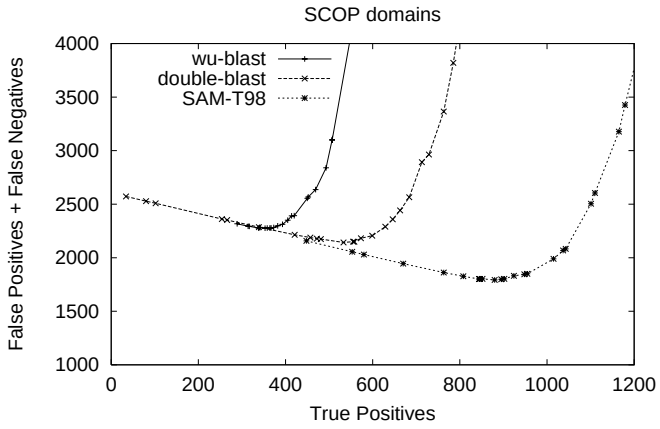
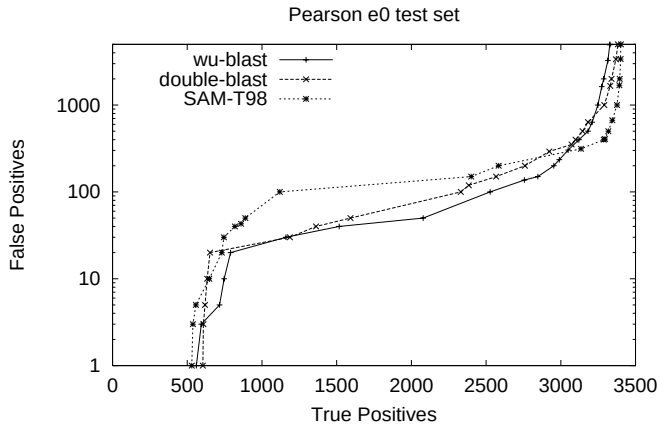


Fig. 7. Results for the methods on the SCOP domain test of 935 sequences. This test set provides the most dramatic evidence of SAM-T98's superior ability over WU-BLASTP and DOUBLE-BLAST as a remote homology detection method. The maximum possible number of true positives is 2605.



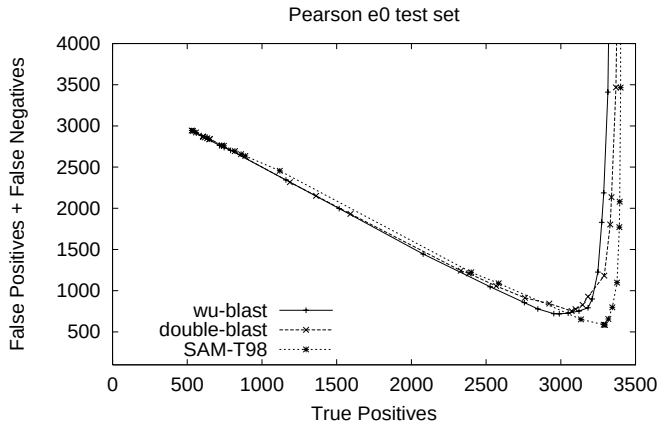


Fig. 8. Results for the methods on the Pearson dataset. The maximum number of true positives is 3474. WU-BLASTP does best for close homologs, and SAM-T98 does best for more remote ones.

(near WU-BLASTP's minimum-error point), WU-BLASTP gets 2952 true positives, DOUBLE-BLAST gets 2760, and SAM-T98 gets 2584. At 400 false positives (near SAM-T98's minimum-error point), WU-BLASTP gets 3121 true positives, DOUBLE-BLAST gets 3099, and SAM-T98 gets 3287. Figure 8 shows this tradeoff in performance clearly. The SAM-T98 method was optimized for finding superfamilies, not families, and so it merges similar families together.

Note that we use a single threshold for each method for all of the targets in a test set, not a separate threshold for each target as done previously for the Pearson test set (Agarwal & States, 1998; Karchin & Hughey, 1998). Using separate thresholds would provide much more impressive numbers, but the single-threshold is a more valuable test. We are not testing how well a particular library of models can be tuned, but how well a set of homologs can be found for a protein of unknown character. If we do not already know the classification, we cannot choose a classification-specific threshold, hence the insistence on a single threshold. If we had used an optimal threshold for each family, the SAM-T98 minimum error point would have dropped from 584 to 285 errors. At this point, there were 3274 true positives and 148 false positives.

4.4 Folds, superfamilies, families, or subfamilies

The SCOP database is a hierarchical classification of protein domain structures, with classification into *class*, *fold*, *superfamily*, *family*, and *subfamily*. We chose to consider pairs that were in the same superfamily to be correct matches, but we could have chosen any level of

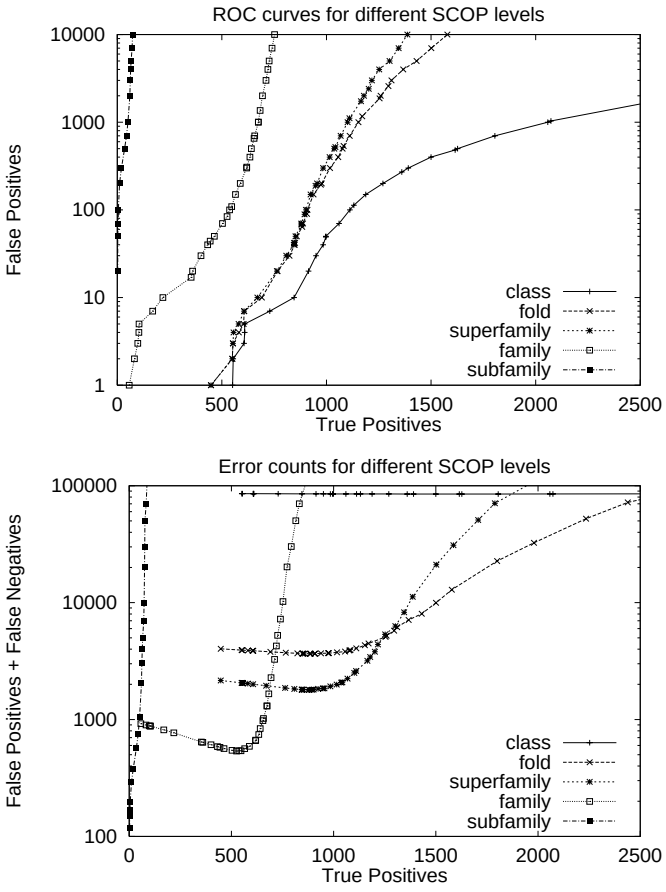


Fig. 9. SAM-T98 results for the SCOP domains test set with correctness defined as matching at different levels of the SCOP hierarchy. The false-positive curves are almost identical for folds and superfamilies at low error rates. (The error plot uses a log scale because of the huge differences in the number of false negatives among the definitions of correctness.)

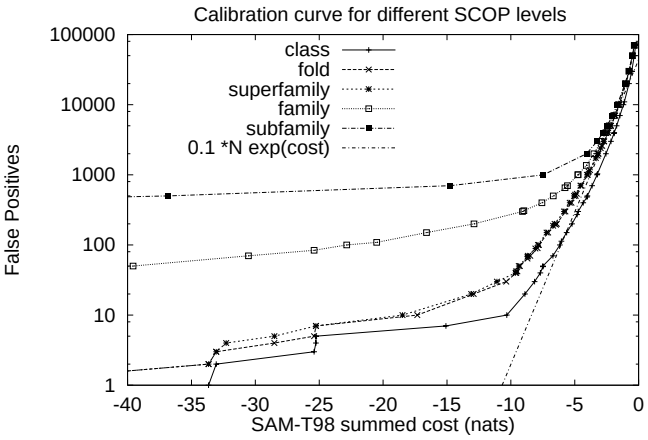


Fig. 10. False positives versus the sum of the two SAM costs (for target and template model), using local alignment and reversed-sequence null model on the SCOP domains test set. The number of false positives does not drop to zero for families or subfamilies because more remote homologs are included in the alignments used to build the HMMs. $N = 436645$ is the number of homology pair tests tried.

the hierarchy as our definition of correctness. Figure 9 shows how choosing different levels would affect our results for the SAM-T98 method. The almost identical false-positive rates for folds and superfamilies at low error rates means that overall error rate is much lower for superfamilies than for folds since there are many more false negatives at the fold level.

The SAM-T98 method also seems to do well on families in Figure 9, but a closer look at the calibration curve in Figure 10 shows that the homologs included in the SAM-T98 alignments are distant enough to contaminate the method as a family or subfamily recognizer (as was seen with the Pearson test set). We would have to use stricter thresholds in building the alignments to create a family, rather than a superfamily, recognizer.

The calibration curve in Figure 10 can be applied to a search with one target in a database of N sequences, the number of false positives from the curve should be multiplied by $N/436,645$ to get the expected number of false positives.

If one ignores the "fat tail" (the excessive number of false positives for strong scores), the number of false positives can be reasonably approximated by $0.1Ne^{\text{cost}}$. The fat tail probably results from two sources of error: small shared motifs (such as ampipathic helices) that are not long enough to justify classifying the proteins in the same superfamily and contamination of the SAM-T98 alignment by non-homologous sequences.

5 Discussion

We have introduced the SAM-T98 method for remote protein homolog detection and have compared it with more popular methods using four test sets. At the minimum-error points, the best method was always SAM-T98. If we evaluate the methods according to the fraction of the possible true homologs found at the minimum error point, SAM-T98 finds 18.8% for the FSSP test set, 42.6% for the SCOP whole chains, 33.8% for the SCOP domains, and 94.9% for the Pearson test set. Even the best current remote-homology method finds only a small fraction of the evolutionary relationships available.

method	FSSP	SCOP whole chain	SCOP domain	Pearson
optimum, true +	2449.45	931	2605	3474
optimum, false +	433.55	0	0	0
optimum, errors	1494.95	0	0	0
WU-BLASTP, true +	173.75	212	353	2948
WU-BLASTP, false +	26.25	21	24	195
WU-BLASTP, errors	3363.35	740	2276	721
double-blast, true +	279.30	288	533	3072
double-blast, false +	50.00	22	71	352
double-blast, errors	3281.55	665	2143	754
target-SAM-T98, true +	421.23	338	869	3296
target-SAM-T98, false +	79.78	15	72	406
target-SAM-T98, errors	3169.40	557	1808	584
SAM-T98, true +	459.68	397	880	–
SAM-T98, false +	81.33	21	68	–
SAM-T98, errors	3132.50	555	1793	–

Table I. Table of minimum-error points for the different test sets and different methods. Each column reports the number of true positives, false positives, and errors (false positives plus false negatives) for one of the four test sets. Target-SAM-T98 refers to the direction of scoring in which an HMM is built for the target sequence and used to score the library of sequences, as opposed to scoring the target sequence with the HMMs built for the library sequences. The SAM-T98 results were generated using the sum of the the scores from both of these directions.

SAM-T98 introduced reversed-model score adjustment. Not only does this scoring method correct for length and composition biases, but some other, subtler effects are also cancelled—for example the periodic hydrophobicity patterns of amphipathic helices or beta strands also appear in the reversed sequence, as does the lower frequency surface-core hydrophobicity pattern. Because of these subtle effects, the reversed sequence is a much more realistic decoy than a scrambled sequence.

These effects can affect scoring signifcantly. For example, in the scoring for the CASP-2 contest (Karplus *et al.*, 1997), we had to eliminate by hand some coiled-coil models that scored any helical protein well—the reversed-model scoring eliminates these problems. Also, metallothionein (4mt2), with 24 cysteines out of 61 residues, can align well to almost any sequence with conserved cysteines. Since many HMMs get a large part of their score from aligning highly conserved cysteines, 4mt2 often appeared as a false postive for these HMMs, but since the reversal of 4mt2 has the same number of cysteines with the same distribution of spacings, it also scores well for these HMMs and the difference between the model and reversed-model scores is near zero.

The SAM-T98 method also introduced score summing. We performed score summing for CASP-2 also—what is new here is the systematic evaluation of this approach. Summing entails the added computational expense of building additional HMMs that is not always clearly justified. For the SCOP whole-chain and domain test sets, summing of scores provided neglible gain in performance. This was not the case for the FSSP test set, for which summing provided a marked improvement for homologs more remote than the minimum error point.

If score summing is to be used, then models must be built for both the target sequences and the database sequences. If not, then only database HMMs or target sequence HMMs are built. Which HMMs to build depends on the number of sequences to classify and the number of families to classify into. If only a small number of sequences are to be identified, then it is probably better to build an HMM for each. If many are to be classified, then building a library of models is better.

The SAM-T98 method uses reversed-model scoring and score thresholds for selecting training sequences in its iterative procedure. We found that a predecessor method (SAM-T97) that did not include the reversed-model scoring and used more liberal threshold values garnered more remote homologs at the expense of including more spurious sequences in the alignments. This led to the creation of HMMs of a slightly different character; they were often more adept at finding the remoter homologies but not as able at filtering out false positives. This is illustrated in Figure 11. While we believe that reversed-model scoring should be retained, we will be in-

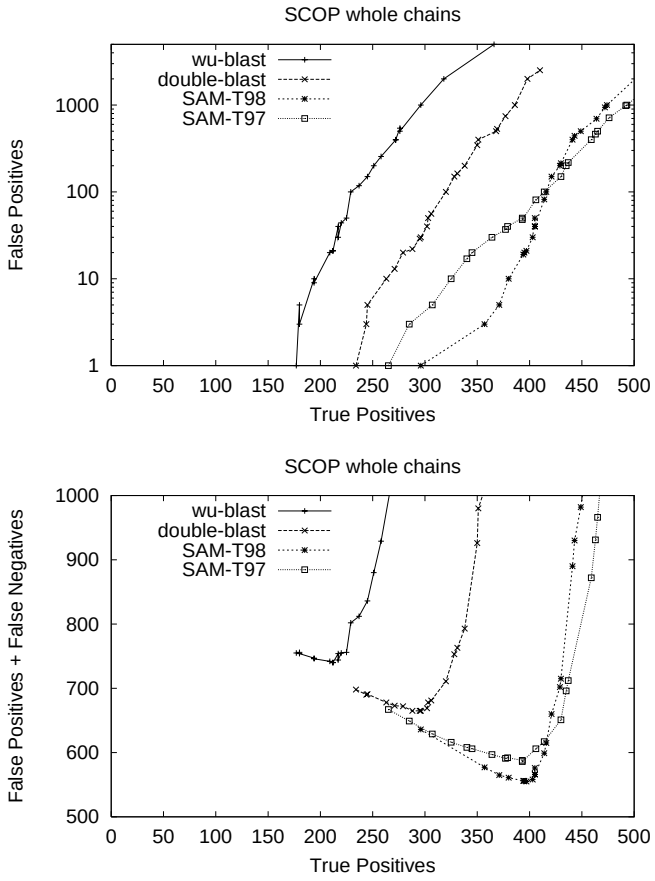


Fig. 11. HMMs with different recognition abilities can be created by adjusting the iterative procedure of SAM-T98. Here, "SAM-T97" refers to a predecessor method that lacked the reversed-model scoring and used more liberal score thresholds.

vestigating the proper threshold settings to find the best balance for building both sensitive and accurate HMMs.

The SAM-T98 method and its use as exemplified in this work is available on the World Wide Web from `http://www.cse.ucsc.edu/research/compbio/`. One may search large sequence databases for homologs using a single query sequence. This makes use of the SAM-T98 method to build an HMM and search a database. Since it is not feasible to build HMMs for all database sequences, database scoring does not sum any scores. The second option allows one to search our model library with a sequence. This is similar to the first option, except that the database is composed of selected sequences from PDB. Since we have constructed HMMs for each of these sequences, score summing is used. Other options allow access to separate components of the SAM-T98 method. They allow one to build an alignment from a query sequence, generate sequence weights from an alignment, or build an HMM from a single query sequence or an alignment with weights.

Future work is needed in several directions: evaluating other fold-recognition methods, tuning the parameters (such as thresholds and number of iterations) of SAM-T98, and evaluating the quality of alignments produced as a by-product of the fold recognition. Other fold-recognition techniques that need to be evaluated include other sequence-based methods for finding relationships, such as MetaMEME (Grundy *et al.*, 1997) and SearchWise (Birney *et al.*, 1996), structure-structure comparison techniques, and methods such as threading that use structure information for the template sequence, but not the target sequence. Some of the more popular sequence-based methods, including PSI-BLAST (Altschul *et al.*, 1997) and ISS (Park *et al.*, 1997), have already been tested on the SCOP dataset (Park *et al.*, 1998). One attempt at testing structure-structure aligners has been made (Gerstein & Levitt, 1998), but that experiment looked only at pairs known to be in the same superfamily, so no false-positive rate can be determined.

Acknowledgements

We thank Nguyet Manh, who ran many of the early tests of the SAM-T97 method on the FSSP test set and did much of the work making SAM-T98 available on the web site, and Cyrus Chothia, who provided us with the selected sequences that make up the SCOP test sets. Special thanks to Philipp Bucher and Kay Hofmann, whose use of reversed sequence databases for normalizing HMMs inspired our somewhat different use of reversed model scoring.

This work was supported in part by NSF grant DBI-9408579, DOE grant DE-FG0395ER62112, and a grant from Digital Equipment Corporation.

References

Agarwal, P. & States, D. J. (1998). Comparative accuracy of methods for protein-sequence similarity search. *Bioinformatics,* **14** (1), 40–47.

Altschul, S., Madden, T., Schaffer, A., Zhang, J., Zhang, Z., W., M., & D., L. (1997). Gapped BLAST and PSI-BLAST: A new generation of protein database search programs. *NAR,* **25**, 3899–3402.

Altschul, S. F. (1991). Amino acid substitution matrices from an information theoretic perspective. *JMB,* **219**, 555–565.

Altshul, S. F., Gish, W., Miller, W., W., M. E., & J., L. D. (1990). Basic local alignment search tool. *JMB,* **215**, 403–410.

Baldi, P., Chauvin, Y., Hunkapillar, T., & McClure, M. (1994). Hidden Markov models of biological primary sequence information. *PNAS,* **91**, 1059–1063.

Barker, W., George, D., & Hunt, L. (1990). Protein sequence database. *Methods Enzymol.* **183**, 31–49.

Barrett, C., Hughey, R., & Karplus, K. (1997). Scoring hidden Markov models. *CABIOS,* **13** (2), 191–199.

Birney, E., Thompson, J., & Gibson, T. (1996). PairWise and SearchWise: finding the optimal alignment is a simultaneous comparison of a protein profile against all DNA translation frames. *NAR,* **24**, 2730–2739.

Brenner, S. E. (1996). *Molecular propinquity: evolutionary and structural relationships of proteins.* PhD thesis University of Cambridge Cambridge, England.

Bucher, P. & Bairoch, A. (1994). A generalized profile syntax for biomolecular sequence motifs and its function in automatic sequence interpretation. In: *ISMB-94* pp. 53–61, Menlo Park, CA: AAAI/MIT Press.

Bucher, P., Karplus, K., Moeri, N., & Hoffman, K. (1996). A flexible motif search technique based on generalized profiles. *Computers and Chemistry,* **20** (1), 3–24.

Eddy, S. (1995). Multiple alignment using hidden Markov models. In: *ISMB-95*, (Rallings, C. *et al.*, eds) pp. 114–120, Menlo Park, CA: AAAI/MIT Press.

Eddy, S., Mitchison, G., & Durbin, R. (1995). Maximum discrimination hidden Markov models of sequence consensus. *J. Comput. Biol.* **2**, 9–23.

Gerstein, M. & Levitt, M. (1998). Comprehensive assessment of automatic structural alignment against a manual standard, the SCOP classification of proteins. *Protein Sci.* **7**, 445–456.

Grundy, W. N., Bailey, W., Elkan, T., & Baker, C. (1997). Meta-MEME: Motif-based hidden Markov models of protein families. *CABIOS,* **13** (4), 397–406.

Haussler, D., Krogh, A., Mian, I. S., & Sjölander, K. (1993). Protein modeling using hidden Markov models: Analysis of globins. In: *Proceedings of the Hawaii International Conference on System Sciences* volume 1 pp. 792–802, Los Alamitos, CA: IEEE Computer Society Press.

Henikoff, S. & Henikoff, J. G. (1994). Position-based sequence weights. *JMB,* **243** (4), 574–578.

Holm, L. & Sander, C. (1993). Protein structure comparison by alignment of distance matrices. *JMB,* **233** (1), 123–138.

Holm, L. & Sander, C. (1996). The FSSP database: Fold classification based on structure-structure alignment of proteins. *NAR,* **24** (1), 206–209.

Holm, L. & Sander, C. (1997). Dali/FSSP classification of three-dimensional protein folds. *NAR,* **25**, 231–234.

Hubbard, T., Murzin, A., Brenner, S., & Chothia, C. (1997). SCOP: a structural classification of proteins database. *NAR,* **25** (1), 236–9.

Hughey, R. & Krogh, A. (1996). Hidden Markov models for sequence analysis: Extension and analysis of the basic method. *CABIOS,* **12** (2), 95–107. Information on obtaining SAM is available at `http://www.cse.ucsc.edu/research/compbio/sam.html`.

Karchin, R. & Hughey, R. (1998). Weighting hidden Markov models for maximum discrimination. Bioinformatics to appear.

Karplus, K., Kimmen Sjölander, Barrett, C., Cline, M., Haussler, D., Hughey, R., Holm, L., & Sander, C. (1997). Predicting protein structure using hidden Markov models. *Proteins: Structure, Function, and Genetics,* **Suppl. 1**, 134–139.

Krogh, A., Brown, M., Mian, I. S., Sjölander, K., & Haussler, D. (1994). Hidden Markov models in computational biology: Applications to protein modeling. *JMB,* **235**, 1501–1531.

McClure, M., Smith, C., & Elton, P. (1996). Parameterization studies for the SAM and HMMER methods of hidden Markov model generation. In: *ISMB-96* pp. 155–164, St. Louis: AAAI Press.

NRP (1998). NRP (Non-Redundant Protein) Database. Distributed on the Internet via anonymous FTP from ftp.ncifcrf.gov, under the auspices of the National Cancer Institute's Frederick Biomedical Supercomputing Center.

Park, J., Karplus, K., Barrett, C., Hughey, R., Haussler, D., Hubbard, T., & Chothia, C. (1998). Sequence comparisons using multiple sequences detect twice as many remote homologues as pairwise methods. *JMB,* **to appear**. http://cyrah.med.harvard.edu/assess_final.html.

Park, J., Teichmann, S., Hubbard, T., & Chothia, C. (1997). Intermediate sequences increase the detection of homology between sequences. *JMB,* **273**, 349–354.

Pearson, W. (1995). Comparison of methods for searching protein sequence databases. *Protein Science,* **4**, 1145–1160.

Pearson, W. & Lipman, D. (1988). Improved tools for biological sequence comparison. *Proc. Natl. Acad. Sci. USA,* **85**, 2444–2448.

Sjölander, K., Karplus, K., Brown, M. P., Hughey, R., Krogh, A., Mian, I. S., & Haussler, D. (1996). Dirichlet mixtures: A method for improving detection of weak but significant protein sequence homology. *CABIOS,* **12** (4), 327–345.

Smith, T. F. & Waterman, M. S. (1981). Identification of common molecular subsequences. *JMB,* **147**, 195–197.

Sonnhammer, E., Eddy, S., & Durbin, R. (1997). Pfam: A comprehensive database of protein families based on seed alignments. *Proteins,* **28**, 405–420.

Tarnas, C. & Hughey, R. (1998). Reduced space hidden Markov model training. *Bioinformatics,* **14** (5), 401–406.