BIOINFORMATICS ORIGINAL PAPER

Vol. 21 no. 12 2005, pages 2827–2831
doi:10.1093/bioinformatics/bti433

Sequence analysis

Convergent Island Statistics: a fast method for determining local alignment score significance

Aleksandar Poleksic*, Joseph F. Danzer, Kevin Hambly and Derek A. Debe

Eidogen-Sertanty Inc., 9381 Judicial Dr., San Diego, CA 92121, USA

Received on March 2, 2005; revised and accepted on April 4, 2005
Advance Access publication April 7, 2005

ABSTRACT

Motivation: Background distribution statistics for profile-based sequence alignment algorithms cannot be calculated analytically, and hence such algorithms must resort to measuring the significance of an alignment score by assessing its location among a distribution of background alignment scores. The Gumbel parameters that describe this background distribution are usually pre-computed for a limited number of scoring systems, gap schemes, and sequence lengths and compositions. The use of such look-ups is known to introduce errors, which compromise the significance assessment of a remote homology relationship. One solution is to estimate the background distribution for each pair of interest by generating a large number of sequence shuffles and use the distribution of their scores to approximate the parameters of the underlying extreme value distribution. This is computationally very expensive, as a large number of shuffles are needed to precisely estimate the score statistics.

Results: Convergent Island Statistics (CIS) is a computationally efficient solution to the problem of calculating the Gumbel distribution parameters for an arbitrary pair of sequences and an arbitrary set of gap and scoring schemes. The basic idea behind our method is to recognize the lack of similarity for any pair of sequences early in the shuffling process and thus save on the search time. The method is particularly useful in the context of profile–profile alignment algorithms where the normalization of alignment scores has traditionally been a challenging task.

Contact: aleksandar@eidogen.com

Supplementary information: http://www.eidogen-sertanty.com/Documents/convergent_island_stats_sup.pdf

INTRODUCTION

Sequence homology detection algorithms employ rare event statistical approaches to estimate the significance of an alignment between two gene or protein sequences. For ungapped local alignments and in the asymptotic limit of long sequences, it is well established that the alignment scores follow an extreme value distribution described by two parameters λ and K (Altschul *et al*., 1990, 1997; Dembo *et al*., 1994). For profile-based algorithms, with or without gaps, it has been conjectured that the score distribution is still of the Gumbel form. However, estimating alignment score significance based on a single or a finite set of Gumbel parameters can introduce errors. In practice λ may vary by >10% from one pair of sequences to another, due to variations in sequence composition (Altschul *et al*., 2001). On the other hand, for marginally significant alignments, even a 4% error in λ leads to an error in E-value greater than a factor of 2.7 (Altschul *et al*., 2001). Various approaches to addressing sequence specific features during the score normalization have been used in lieu of a rigorous theory. In PSI-BLAST, lengths are dealt with using edge-effect correction (Altschul and Gish, 1996) and composition-based effects are dealt with using composition-based statistics (Schaffer *et al*., 2001). There are other efficient methods not implemented in BLAST (Mott and Tribe, 1999; Mott, 2000) that estimate length- and composition-dependent statistical parameters without further simulation.

In contrast to profile-sequence methods, profile–profile alignment algorithms still lack a fast and accurate assessment of the score statistics. This is in particular true for algorithms that use structural information, position specific gaps and various other constraints in the alignment process. Accurate score normalization in these methods is very important because there is plenty of evidence suggesting that profile–profile algorithms can recognize some extremely distant sequence relationships. In the last CASP experiment, for example, profile–profile methods were among the top performers across all categories.

Many of the existing profile–profile alignment methods use Z-score statistics to measure the score significance (Rychlewski *et al*., 2000; Ginalski *et al*., 2003). Other methods normalize the alignment scores by assessing their locations in a fixed Gumbel distribution. The first approach is limited in the number of shuffles that can be employed in order to process the database in a timely fashion. It also makes the wrong assumption about the Gaussian form of the underlying score distribution. The second technique is extremely fast, but it does not address the profiles' lengths and the composition bias. COMPASS (Sadreyev and Grishin, 2003) generalizes the PSI-BLAST approach in computing alignment score significance. However, the method is designed to work with the COMPASS alignment algorithm, i.e. for its specific scoring function and gap penalties.

Recently, methods that use scores for local alignment 'islands' have been described (Olsen *et al*., 1999; Altschul *et al*., 2001). The so-called 'island method' overcomes many of the bottlenecks of the earlier methods. But, despite its favorable properties, it alone still does not allow for quick estimates of λ and K in the context of a large database search. In the island method, independent of how 'similar' sequences are, they are repeatedly shuffled and aligned in order to produce a large number of island scores needed to derive the two Gumbel parameters.

Our method, Convergent Island Statistics (CIS), builds upon the island method, but is able to recognize the lack of sequence similarity

*To whom correspondence should be addressed.

early in the shuffling process and thus save on the search time. Full blown shuffling is needed only when there is enough evidence that the sequences are related. Convergent Island Statistics has the following properties:

(1) It is reasonably fast. Profile–profile alignment algorithms incorporating CIS are able to process standard databases like PFAM or PDB in real time.

(2) Since it is based on sequence shuffles, the estimation of distribution parameters in CIS takes account of sequence (profile) lengths and compositions.

(3) CIS provides an explicit, analytical tradeoff between the algorithm's speed and sensitivity.

(4) It uses no lookup tables and contains no parameters to optimize.

(5) It estimates score statistics 'on-the-fly' and therefore it can be readily applied to a broad class of local alignment algorithms, without pre-processing of the data of any kind.

Background

Statistics of sequence alignment scores have been extensively studied (Dembo *et al*., 1994; Altschul and Gish, 1996; Mott and Tribe, 1999; Mott, 2000; Karlin and Altschul, 1990, 1993; Pearson, 1998; Collins *et al*., 1988; Mott, 1992; Waterman and Vingron, 1994a,b). For sequence–sequence alignments lacking gaps, the expected number of locally optimal sub-alignments with a score of at least x is approximately Poisson distributed with mean value E,

$$E = Kmne^{-\lambda x} \tag{1}$$

In Equation (1), m and n are the lengths of the sequences, K is the natural scale for the size of the search space and λ is the scale parameter for the scoring system. Equation (1) implies that the probability of finding exactly k alignments with score $\geq x$ is

$$e^{k \ln E - E}/k! \tag{2}$$

Hence, the probability of finding at least one such alignment is

$$P(S \geq x) = 1 - e^{-E} \tag{3}$$

The last quantity is called the P-value of the score x, and is the measure of statistical significance of x. From now on, we will denote the P-value of the score x by $p(x|\lambda, K)$ in order to specify the underlying distribution parameters. Note that the accurate estimates of λ are much more important than those of K, since λ enters Equation (1) exponentially.

Island Statistics

Recently, Olsen *et al*. (1999) proposed a computationally efficient method for determining λ and K using scores for local alignment 'islands'. The value of each cell in a Smith–Waterman matrix corresponds to the highest scoring local alignment that ends at that cell. The local alignment starts at a so-called anchor cell, and an island is defined to be the set of all cells that have the same anchor. The score of an island is the maximum score of the cells it contains. A simple modification of the Smith–Waterman algorithm involving a small extra computation per cell allows one to keep track of the anchor cells, as well as the island end-points and their scores. Since island scores are scores of distinct sub-optimal alignments, Equation (1) describes well the number of islands with a score of at least x, and is increasingly accurate for larger values of x. Thus, the precise estimates of λ and K may be obtained by considering those islands with scores of at least some threshold value c. For the case of discrete alignment scores, the maximum likelihood estimate of λ is

$$\hat{\lambda} = \ln\left(1 + \frac{1}{S_c}\right) \tag{4}$$

where

$$S_c = \frac{1}{N}\sum_{i \in I_c}(S(i) - c) \tag{5}$$

and where $S(i)$ is the score of the ith island, $I_c = \{i|S(i) \geq c\}$ and $N = |I_c|$ (Altschul *et al*., 2001).
In the case of continuous alignment scores,

$$\hat{\lambda} = \frac{1}{S_c} \tag{6}$$

The maximum likelihood estimate of K is

$$\hat{K} = \frac{Ne^{\hat{\lambda}_c c}}{A} \tag{7}$$

where A is the aggregate search area of the island search space (Altschul *et al*., 2001). For example, if two sequences of lengths m and n were compared once, $A = mn$. If P such comparisons were made then $A = Pmn$ (Altschul *et al*., 2001). For the sake of simplicity, hereafter we will focus on continuous alignment scores and use $\hat{\lambda} = 1/S_c$.

SYSTEMS AND METHODS

It has been shown that the island method has a speed advantage over the direct shuffling method. For recommended asymptotic parameter estimation, the speed advantage of the island method even approaches an order of magnitude (Altschul *et al*., 2001). Yet, in the context of a database search it is still too time-consuming to re-estimate λ and K for each sequence pair of potential interest. One accurate estimate of the two parameters would require as much time as searching a typical current database with a standard heuristic method.

Convergent Island Statistics is capable of quickly recognizing significant matches and estimating the score distribution only in the case where there is evidence that the two sequences are related. Although our method is applicable to a more general setting, for the sake of simplicity we will only describe the version that is an enhancement of the island statistics method. The main idea is based upon the following two observations:

(1) If $\lambda_1 \leq \lambda_2$ and $K_1 \leq K_2$ then $p(x|\lambda_2, K_1) \leq p(x|\lambda_1, K_2)$.

(2) The distribution of $\hat{\lambda}/\lambda$ is approximately normal with mean 1 and standard deviation $1/\sqrt{N}$. The distribution of $\hat{K}/K$ is approximately normal with mean 1 and standard deviation $1/\sqrt{KNmne^{-\lambda c}}$ (see the Supplementary material).

Note that observation 1 implies that if an alignment score is not statistically significant with respect to the extreme value distribution EVD_{λ_2,K_1}, then it is not statistically significant with respect to EVD_{λ_1,K_2} for any $\lambda_1 \leq \lambda_2$ and $K_1 \leq K_2$. On the other hand, observation 2 allows one to estimate and control the probability that the true λ is within an interval that depends on sampled $\hat{\lambda}$ and the number of islands used to estimate $\hat{\lambda}$. The same argument can be applied to the other parameter K. Observation 2 further implies that the confidence intervals corresponding to the jth multiple of the standard deviation σ are respectively

$$1 - j\sigma < \frac{\hat{\lambda}}{\lambda} < 1 + j\sigma \tag{8}$$

and

$$1 - j\sigma < \frac{\hat{K}}{K} < 1 + j\sigma \tag{9}$$

These inequalities are equivalent to

$$\lambda(j)_N^- < \lambda < \lambda(j)_N^+ \tag{10}$$

where

$$\lambda(j)_N^- = \frac{\hat{\lambda}}{1 + j/\sqrt{N}} \quad \text{and} \quad \lambda(j)_N^+ = \frac{\hat{\lambda}}{1 - j/\sqrt{N}}$$

and

$$K(\lambda, c)_N^- < K < K(\lambda, c)_N^+ \tag{11}$$

where

$$K(\lambda, c)_N^- = \hat{K} - \frac{je^{\lambda c}}{\sqrt{Nmn}} \quad \text{and} \quad K(\lambda, c)_N^+ = \hat{K} + \frac{je^{\lambda c}}{\sqrt{Nmn}}$$

ALGORITHM

Suppose that we are given a sequence S_q and we want to search a (possibly large) database $\{S_t^1, \ldots, S_t^R\}$ for sequences similar to S_q. Assume that we are only interested in the matches with the P-value below a certain cutoff p_0 and we want to precisely estimate the P-value for every such match. For each target sequence, our algorithm is a series of steps $s_1, \ldots, s_k$, where each step s_i may be described as follows.

Shuffle the sequences (or columns of both sequential profiles) as many times as needed to generate N_i islands. Note that shuffling randomizes the order of the symbols in a sequence without changing the sequence's composition. Calculate $\lambda(j)_{N_i}^+$ and $K(\lambda(j)_{N_i}^+)_{N_i}^-$. If $p(x|\lambda(j)_{N_i}^+, K(\lambda(j)_{N_i}^+)_{N_i}^-) > p_0$ (according to observation 1) discard the score as insignificant and proceed to the next sequence. Otherwise, go to the next step. If reached, the last step k ends with computing the final P-value from the distribution with parameters $\hat{\lambda}$ and $\hat{K}$ which are estimated from N_k islands.

The pseudo-code for the CIS algorithm is given below.

```
for r := 1 to R do
    for i := 1 to k do
        while(number of islands < $N_i$)
            shuffle $S_q$
            shuffle $S_t^r$
            align shuffled sequences and extract more islands;
        end-while
        compute $\lambda(j)_{N_i}^+$, and $K(\lambda(j)_{N_i}^+)_{N_i}^-$
        Pval :=$p(x|\ \lambda\ (j)_{N_i}^+, K(\lambda\ (j)_{N_i}^+)_{N_i}^-)$
        if Pval > $p_0$ then
            print 'No similarity found.'
            exit inner for-loop (go to the next sequence)
        else if i == k
            print alignment between $S_q$ and $S_t^r$
            print Pval.
        end-if
    end-for
end-for
```

The number of islands N_i in our algorithm increases at each step ($N_1 < \cdots < N_k$). This allows for more precise estimates of the parameters compared to the estimates in the previous step, but it also requires more CPU time.

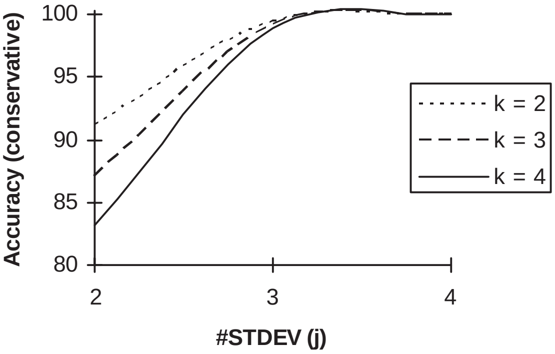

Fig. 1. The algorithm's accuracy as a function of j (#STDEV) and k (#STEPS).

Analysis

The algorithm's accuracy is defined as the percentage of true positives (i.e. those found by the regular island method to have the P-value $< p_0$) that are not discarded in any of the algorithm's steps. The accuracy depends on both j (the number of standard deviations above the mean in the distributions of $\hat{\lambda}/\lambda$ and $\hat{K}/K$) and k (the number of algorithm steps). For example, $j = 3$ corresponds to ~99.86% one-tailed confidence interval in the Gaussian distribution, which means that Equations (8) and (9) are each true ~99.86% of the time. But, since both $\hat{\lambda}$ and $\hat{K}$ are required to pass the tests, assuming (wrongly but conservatively) independence of $\hat{\lambda}$ and $\hat{K}$, the confidence drops to about 99.7% (in other words, the chances of keeping a significant hit in any pass of the algorithm are at least 99.7%). Also, $\hat{\lambda}$ and $\hat{K}$ have to pass the test k separate times. Assuming (again wrongly but conservatively) independence of the k tests, the confidence decreases accordingly. For $j = 3$ and $k = 3$, the confidence interval is at least 99.1%. Figure 1 shows the relationship between j, k and the accuracy of the parameter estimates.

In the case of BLOSUM62 substitution matrix with affine gap scores of $-(11 + k)$ for gaps of length k, the best cutoff score for saving the alignment islands is around $c = 28$ (Altschul *et al.*, 2001). In general, for an arbitrary alignment algorithm (including profile-based methods), a cutoff score of $c = const * m$ can be used, where m is the largest positive entry of the scoring matrix (Olsen *et al.*, 1999).

As in the regular island method, the estimates of statistical parameters in CIS can be made free of edge-effect bias by extending the dynamic programming matrix and considering only those islands that are anchored within the central region of the matrix (Altschul *et al.*, 2001).

To test the effectiveness of our method, we have implemented a simple version of the Smith–Waterman algorithm and compared the performance of CIS and the regular island method on Lindahl's dataset (Lindahl and Elofssons, 2000). The algorithm uses the BLOSUM62 substitution matrix in conjunction with affine gap scores of $-(11 + k)$ for gaps of length k, and the island threshold score of $c = 28$. To reduce edge-effects, each dynamic programming matrix is surrounded by a border of width 100, filled with the randomly chosen scores from the central area of the matrix (Altschul *et al.*, 2001). For the sake of simplicity and consistency, we assume continuous alignment scores. However, the results should be

Table 1. Speed comparison between CIS and the regular island method using P-value cutoff $p_0 = 2e - 6$ (corresponding to E-value of $\sim$1.0)

j	k	Number of islands 1st pass	 2nd pass	 3rd pass	Speed ratio (CIS: island m)	Accuracy
3	2	100	10 000		87.8	99.46%
4	2	100	10 000		84.2	99.99%
3	3	100	400	10 000	92.6	99.19%
4	3	100	400	10 000	92.0	99.98%

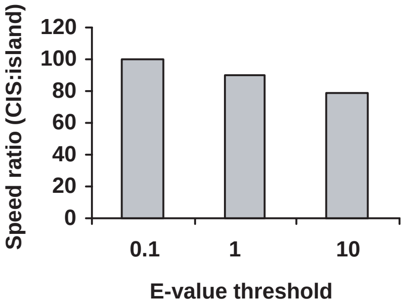

Fig. 2. The speed advantage of CIS over the island method, as a function of E-value threshold E_0. The remaining parameters are set to $j = 4, k = 3$ (100, 400 and 10 000 islands).

comparable to those obtained with discrete alignment scores, as the two distributions of ML estimates are very similar for this algorithm's setting (Altschul *et al*., 2001). The standard error of $\hat{\lambda}/\lambda$ in case of continuous scores is $1/\sqrt{N}$ and the standard error for discrete scores is $(\exp(\lambda) - 1)/\lambda\sqrt{\exp(\lambda)}\sqrt{N} \sim 1.003/\sqrt{N}$ (Altschul *et al*., 2001). In our experiment, the island method is set to generate at least 10 000 islands for each pair of sequences [corresponding to standard error in $\hat{\lambda}/\lambda$ of about 1% (Altschul *et al*., 2001)]. The numbers of islands assigned to various steps of the CIS method are 100, 400 and 10 000, corresponding to standard errors of about 10, 5 and 1%, respectively. The P-value cutoff of $p_0 = 2e - 6$ is chosen to correspond to the E-value cutoff of $E_0 \approx 1$, which is often the default value in database search algorithms. As seen in Table 1, the CIS method is, on average, about 90 times faster then the regular island method.

The actual speed advantage of our method also depends on the P-value cutoff p_0 set in the search. In other words, the speed advantage is higher if one is not interested in weak matches and it is increasing as the P-value cutoff is lowered. Figure 2 shows the speed advantage of CIS over the regular island method using various threshold values for E_0.

It should be noted that the speed will not vary much as the significance threshold is reduced if the database contains many true positives. For example, if a query belongs to an abundantly represented superfamily, CIS will still be slow, as every database hit will be thoroughly evaluated by all steps of the algorithm. Also, in its present form, the algorithm creates a completely new set of islands in each step (i.e. islands from step s_i are not used in step s_{i+1}). Thus, an additional speed gain may be obtained by saving the islands from each pass and using them in subsequent steps of the algorithm.

Table 2. Speed comparison between the regular CIS method and the same method enhanced with filtering of the database hits based on conservative parameter estimates[a]

λ	K	Change in λ (%)	Change in K (%)	Speed ratio (CIS + filtering: CIS)
0.27501	0.04074	+3%	−3%	13.65
0.28035	0.03990	+5%	−5%	12.14
0.29370	0.03780	+10%	−10%	9.02

[a] The true values for λ and K are assumed to be $\lambda = 0.267$ and $K = 0.042$ (Altschul *et al*., 2001). The same parameter setting as in the first example is used.

It is easy to see that the algorithm's speed can be further increased by filtering out the database hits based on a background distribution defined by some fixed, pre-computed, conservative estimates of the parameters λ and K. Table 2 shows the speed comparison between the CIS method described above and the same method applied in conjunction with the filtering of the database hits based on various conservative estimates of λ and K. However, despite the fact that the latter version of CIS has an additional speed advantage over the regular CIS method, this particular approach does not provide 'on-the-fly' statistics, as an additional effort is needed to pre-compute the estimates of the two distribution parameters.

DISCUSSION AND CONCLUSION

In the case of sequence–sequence alignments that are not allowed to contain gaps, the parameters of the score distribution may be calculated analytically. Although no rigorous analytical theory has been developed for profile-based method, the score normalization problem in profile-sequence methods has been extensively studied and efficiently addressed. Profile–profile alignment algorithms still lack fast and accurate score statistics. To account for the rich profile content, structural constraints and different gap models, one has to employ a brute force method of doing extensive random shuffles for every pair of profiles of interest. However, the sequence databases have grown in size enormously over the last few years, making the brute force approach computationally prohibitive.

The method we propose is able to recognize the lack of similarity for any pair of sequences early in the shuffling process and thus save on the search time. Any given sequence will typically have a small number of significant hits in a representative, large database, so the vast percentage of comparisons will be computed very efficiently. On the other hand, if the sequences are related, our method approaches the complexity of the brute force method of doing extensive random shuffles and therefore is able to recognize and precisely estimate the statistics of every such pair.

Convergent Island Statistics can be readily applied to any alignment algorithm whose background scores follow an extreme value distribution. The method contains no parameters to optimize and there is no need for fitting the data of any kind. The CIS method was particularly useful to our group in the preparation for the CASP6 structure prediction experiments (http://predictioncenter.llnl.gov). For CASP6 we needed to test the performance of different profile–profile search strategies by frequently changing the method's parameters, such as gap penalties, scoring schemes and weights on various other input data. Having a method for computing 'on-the-fly' statistics proved to be very convenient. It would be almost impossible to test and validate various theoretical and heuristic approaches to database search if we had to re-estimate the score statistics each time we switch from one algorithm's setting to another. All three of our automated servers in CASP6 and CAFASP4 (http://www.cs.bgu.ac.il/∼dfischer/CAFASP4/alev.html) experiments, namely Eidogen-EXPM, Eidogen-BNMX and Eidogen-SFST, used CIS to estimate the significance of database hits. However, the detailed description of the algorithms as well as their CASP and CAFASP performance is beyond the scope of this paper and will be published elsewhere.

REFERENCES

Altschul,S.F. and Gish,W. (1996) Local alignment statistics. *Methods Enzymol*., **266**, 460–480.

Altschul,S.F. *et al*. (1990) Basic local alignment search tool. *J. Mol. Biol.*, **215**, 403–410.

Altschul,S.F. *et al*. (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res.*, **25**, 3389–3402.

Altschul,S.F. *et al*. (2001) The estimation of statistical parameters for local alignment score distributions. *Nucleic Acids Res.*, **29**, 351–361.

Collins,J.F. *et al*. (1988) The significance of protein sequence similarities. *Comput. Appl. Biosci*., **4**, 67–71.

Dembo,A. *et al*. (1994) Critical phenomena for sequence matching with scoring. *Ann. Prob.*, **22**, 1993–2021.

Ginalski,K. *et al*. (2003) Detection of distant homology using sequence profiles and predicted secondary structure. N*ucleic Acids Res.*, **31**, 3804–3807.

Karlin,S. and Altschul,S.F. (1990) Methods for assessing the statistical significance of molecular sequence features by using general scoring schemes. *Proc. Natl Acad. Sci. USA*, **87**, 2264–2268.

Karlin,S. and Altschul,S.F. (1993) Aplications and statistics for multiple high-scoring segments in molecular sequences. *Proc. Natl Acad. Sci. USA*, **90**, 5873–5877.

Lindahl,E. and Elofsson,A. (2000) Identification of related proteins on family, superfamily and fold level. *J. Mol. Biol.*, **295**, 613–625.

Mood,A.M., Graybill,F.A. and Boes,D.C. *Introduction to the Theory of Statistics*, 3rd ed. McGraw-Hill, 1974.

Mott,R. (2000) Accurate formula for P-values of gapped local sequence and profile alignments. *J. Mol. Biol.*, **300**, 649–659.

Mott,R. (1992) Maximum-likelihood estimation of the statistical distribution of Smith–Waterman local sequence similarity scores. *Bull. Math. Biol*., **54**, 59–75.

Mott,R. and Tribe,R. (1999) Approximate statistics of gapped alignments. *J. Comput. Biol.*, **6**, 91–112.

Olsen,R., Bundschuh,R. and Hwa,T. (1999) Rapid assessment of extremal statistics for gapped local alignment. In: Lengauer,T., Schneider,R., Bork,P., Brutlag,D., Glasgow,J., Mewes,H.-W. and Zimmer,R. (eds), *Proceedings of the Seventh International Conference on Intelligent Systems for Molecular Biology.* AAAI Press, Menlo Park, CA, pp. 211–222.

Pearson,W.R. (1998) Empirical statistical estimates for sequence similarity searches. *J. Mol. Biol.*, **276**, 71–84.

Rychlewski,L. *et al*. (2000) Comparison of sequence profiles. Strategies for structural predictions using sequence information. *Protein Sci.*, **9**, 232–241.

Sadreyev,R.I. and Grishin,N.V. (2003) COMPASS: a tool for comparison of multiple protein alignments with assessment of statistical significance. *J. Mol. Biol.*, **326**, 317–336.

Schaffer,A.A. *et al*. (2001) Improving the accuracy of PSI-BLAST protein database searches with composition-based statistics and other refinements. *Nucleic Acids Res.*, **29**, 2994–3005.

Waterman,M.S. and Vingron,M. (1994a) Sequence comparison significance and Poisson approximation. *Statist. Sci*., **9**, 367–381.

Waterman,M.S. and Vingron,M. (1994b) Rapid and accurate estimates of statistical significance for sequence database searches. *Proc. Natl Acad. Sci. USA*, **91**, 4625–4628.