Multiple sequence alignment using Probcons and Probalign

Usman Roshan

Summary

Sequence alignment remains a fundamental task in bioinformatics. The literature contains programs that employ a host of exact and heuristic strategies available in computer science. Probcons was the first program to construct maximum expected accuracy sequence alignments with hidden Markov models and at the time of its publication achieved the highest accuracies on standard protein multiple alignment benchmarks. Probalign followed this strategy except that it used a partition function approach instead of hidden Markov models. Several programs employing both strategies have been published since then. In this chapter we describe Probcons and Probalign.

Keywords sequence alignment, expected accuracy, hidden Markov models, partition function

1. Introduction

Multiple protein sequence alignment is one of the most commonly used task in bioinformatics [1]. It has widespread applications that include detecting functional regions in proteins [2] and reconstructing complex evolutionary histories [1,3]. Techniques for constructing accurate alignments are therefore of great interest to the bioinformatics community.

ClustalW [4] is one of the earliest multiple sequence aligners and remains popular to date. Other programs include Dialign [5], T-Coffee [6], MUSCLE [7], and MAFFT [8]. Given the importance of multiple sequence alignment, several protein alignment benchmarks have been created for unbiased accuracy assessment of alignment quality. Of these, BAliBASE [9,10,11] is by far the most commonly used. The BAliBASE benchmark alignments are computed using superimposition of protein structures.

Prior to Probcons [12] most programs optimized the sum-of-pairs score of a multiple alignment or computed the Viterbi alignment [3]. Probcons computes the maximal expected accuracy alignment instead. The expected accuracy of an alignment is based upon posterior probabilities of residues [3,12,13,14]. Probcons computes these probabilities using a Hidden Markov Model (HMM) for pairwise sequence alignment. The HMM parameters are learned using unsupervised learning on the BAliBASE 2.0 benchmark.

Probalign [13] on the other hand estimates amino acid posterior probabilities from the partition function of alignments as described by Miyazawa [14]. It then proceeds to compute the maximal expected accuracy multiple sequence alignment by following the strategy of Probcons. We first describe both methods of computing posterior probabilities in detail below. We then describe the Probcons alignment algorithm that makes use of the probabilities to output a final alignment. Probalign follows the same approach.

2. Methods

2.1 Posterior probabilities for expected accuracy sequence alignment

The expected accuracy of an alignment is based upon the posterior probabilities of aligning residues in two sequences. Consider sequences x and y and let a^* be their true alignment. Following the description in Do [12] the posterior probability of residue x_i aligned to y_j in a^* is defined as

<Equation 1> (1)

where A is the set of all alignments of x and y and $1(expr)$ is the indicator function which returns 1 if the expression $expr$ evaluates to true and 0 otherwise. $P(a|x,y)$ represents the probability that alignment a is the true alignment a^*. This can easily be calculated using a pairwise HMM if all the parameters are known (described below). From hereon we represent the posterior probability as $P(x_i \sim y_j)$ with the understanding that it represents the probability of x_i aligned to y_j in the true alignment a^*.

According to equation (1) as long as we have an ensemble of alignments A with their probabilities $P(a|x,y)$ we can compute the posterior probability $P(x_i \sim y_j)$ by summing up the probabilities of alignments where x_i is paired with y_j . Probcons uses hidden Markov models while Probalign uses the partition function of sequence alignments to generate the ensemble.

2.2 Posterior probabilities by hidden Markov models

Probcons uses a basic sequence alignment hidden Markov model (HMM) shown in Figure 1 below.

<Figure 1>

The emission probabilies for the hidden states M, I_x and I_y are given by $p(x_i,y_j)$, $q(x_i)$, and $q(y_j)$ where x_i is the i^{th} residue of sequence x and y_j defined correspondingly. The terms δ and ε represent transition probabilities for gap open and gap extensions. The probability of a sequence alignment under this model is well-defined and the one with the highest probability can be found with the Viterbi algorithm [3]. The posterior probabilities can then be obtained by

<Equation 2>

In the above equation $f(i,j)$ is the sum of all probabilities of all alignments of $x_{1..i}$ and $y_{1..j}$ where $x_{1..i}$ are the first i characters of sequence x and $y_{1..j}$ is defined the same way. The term $b(i,j)$ is the sum of all probabilities of all alignments of $x_{i+1..m}$ and $y_{j+1..n}$ where m and n are the lengths of sequences x and y respectively. And finally $P(x,y)$ is the sum of the probabilities of all alignments of x and y under the model. These can be obtained by modifying the Viterbi algorithm to add instead of taking the max as shown in Durbin [3].

2.3 Posterior probabilities by partition function

Amino acid scoring matrices that are normally used for sequence alignment are represented as log-odds scoring matrices as defined by Dayhoff [15]. The commonly used sum-of-pairs score of an alignment [3] is defined as the sum of residue-residue pairs and residue-gap pairs under an affine penalty scheme.

<Equation 3>

Here T is a constant and set according to the scoring matrix, M_{ij} is the mutation probability of residue i changing to j and f_i and f_j are background frequencies of residues i and j. In fact, it can be shown that any scoring matrix corresponds to a log odds matrix [16,17]. Miyazawa [14] proposed that the probability of alignment $P(a)$ of sequences x and y can be defined as

<Equation 4>

where $S(a)$ is the score of the alignment under the given scoring matrix. In this setting one can then treat the alignment score as negative energy and T as the thermodynamic temperature, similar to what is done in statistical mechanics. Analogous to the statistical mechanical framework Miyazawa [14] defined the partition function of alignments as

<Equation 5>

where A is the set of all alignments of x and y. With the partition function in hand the probability of an alignment a can now be defined as

<Equation 6>

As T approaches infinity all alignments are equally probable, whereas at small values of T only the nearly optimal alignments have the highest probabilities. Thus, the temperature parameter T can be interpreted as a measure of deviation from the optimal alignment.

The alignment partition function can be computed using recursions similar to the Needleman-Wunsch dynamic algorithm. Let Z^M_{ij} represent the partition function of all alignments of $x_{1..i}$ and $y_{1..j}$ ending in x_i paired with y_j, and $S_{ij}(a)$ represent the score of alignment a of $x_{1..i}$ and $y_{1..j}$. According to equation (2)

<Equation 7>

where A_{ij} is the set of all alignments of $x_{1..i}$ and $y_{1..j}$, and $s(x_i,y_j)$ is the score of aligning residue x_i with y_j. The summation in the bracket on the right hand side of the above equation is precisely the partition function of all alignments of $x_{1..i-1}$ and $y_{1..j-1}$. We can thus compute the partition function matrices using standard dynamic programming.

<Equation 8>

Here $s(x_i,y_j)$ represents the score of aligning residue x_i with y_j, g is the gap open penalty, and ext is the gap extension penalty. The matrix Z^M_{ij} represents the partition function of all alignments ending in x_i paired with y_j. Similarly Z^E_{ij} represents the partition function of all alignments in which y_j is aligned to a gap and Z^F_{ij} all alignments in which x_i is aligned to a gap. Boundary conditions and further details can be obtained from Miyazawa [14].

Once the partition function is constructed, the posterior probability of x_i aligned to y_j can be computed as

<Equation 9>

where $Z'^{M}_{i,j}$ is the partition function of alignments of subsequences $x_{i..m}$ and $y_{j..n}$ beginning with x_i paired with y_j and m and n are lengths of x and y respectively. This can be computed using standard backward recursion formulas [3]. In the above equation $Z^{M}_{i-1,j-1}/Z$ and $Z'^{M}_{i+1,j+1}/Z$ represent the probabilities of feasible suboptimal alignments (as determined by the T parameter) of $x_{1..i-1}$ and $y_{1..j-1}$, and $x_{i+1..m}$ and $y_{j+1..n}$ respectively, where m and n are lengths of x and y respectively. Thus, the equation weighs alignments according to their partition function probabilities and estimates $P(x_i \sim y_j)$ as the sum of probabilities of all alignments where x_i is paired with y_j.

2.4 Maximal expected accuracy alignment

Given the posterior probability matrix $P(x_i \sim y_j)$, we define the expected accuracy of the alignment of x and y as

<Equation 10>

The maximum expected accuracy alignment score is computed by dynamic programming using the following recurrence described in Durbin [3].

<Equation 11>

The first row and column of A are set to zero. The alignment score is given by $A(|x|,|y|)$ where $|x|$ and $|y|$ denote the lengths of sequences x and y. The actual alignment of x and y can be recovered by keeping track of which cell the maximum value is obtained from for each entry of A [3].

Both Probcons and Probalign first estimate posterior probabilities for amino acid residues for all pairs of protein sequences in the input. Probcons introduced a number of new approaches for constructing a multiple alignment with posterior probabilities for all pairs of sequences. It first performs a probabilistic consistency transformation to improve posterior probabilities with the aid of a third sequence [12]. It then adapts three standard approaches in multiple sequence alignment, namely construction of a guide-tree, progressive alignment, and iterative refinement to the expected accuracy alignment approach. The guide-tree construction is similar to UPGMA [18] except that expected accuracies are used to measure distance between clusters [12]. Profile-profile alignment [3], another standard technique in multiple sequence alignment, is extended to incorporate expected accuracies which facilitates the progressive and iterative alignment strategies. Probalign follows all of these procedures for constructing its multiple alignment.

3. Practical Issues

Probalign is freely available at http://probalign.njit.edu [19] with gap penalties optimized for standard protein and RNA alignment benchmarks and Probcons is available from its authors. In

terms of running time both Probcons and Probalign are slower than several previous approaches and so the alignment of thousands of sequences remains a challenge. Some runtime improvements have been made to Probalign and the most recent version 1.4 (at the time of writing this chapter) is considerably faster than earlier ones.

The Probalign webserver, also called eProbalign, provides a useful tool for eliminating poorly aligned columns. The problem of determining reliably aligned columns frequently comes up in practice. eProbalign provides one solution by averaging pairwise posterior probabilities in each column and displaying them in different shades of red. The server also allows the alignment to be saved in text and pdf formats.

In practice Probalign outperforms existing programs by large margins when the data contains sequences of varying lengths [13]. Thus it is particularly suitable for protein and RNA datasets where the sequence length variation is high.

The alignment of genomic length DNA sequences pose a runtime challenge to Probalign and Probcons. Both work best for protein and RNA sequences. However, the program Pecan [20] and webserver plastrna.njit.edu [21] adapt the expected accuracy approach for genome analysis. The former is for genome alignment while the latter searches for evolutionary related RNAs in genomes.

4. References

[1] C. Notredame, (2002) Recent progresses in multiple sequence alignment: a survey, *Pharmacogenomics* 3(1) pp:131-144.

[2] D. La, B. Sutch, and D.R. Livesay, (2005) Predicting protein functional sites with phylogenetic motifs, *Proteins* 58 pp:309-320.

[3] R. Durbin, S. Eddy, A. Krogh, and G. Mitchison, (1998) Biological sequence analysis: probabilistic models of proteins and nucleic acids, Cambridge University Press

[4] J. D. Thompson, D. G. Higgins, and T. J. Gibson, (1994) ClustalW: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position specific gap penalties, and weight matrix choice. *Nucleic Acids Research* 27(13) pp:2682-2690.

[5] A. R. Subramanian, J. Weyer-Menkhoff, M. Kaufmann, and B. Morgenstern, (2005) Dialign-T: an improved algorithm for segment-based multiple sequence alignment, *BMC Bioinformatics* 6 pp:66.

[6] C. Notredame, D. Higgins, and J. Heringa, (2000) T-Coffee: a novel method for multiple sequence alignments, *Journal of Molecular Biology* 302 pp:205-217.

[7] R. C. Edgar, (2004) MUSCLE: multiple sequence alignment with high accuracy and high throughput, *Nucleic Acids Research* 32(5) pp:1792-1797.

[8] K. Katoh, K. Misawa, K. Kuma, and T. Miyata, (2005) MAFFT version 5: improvement in accuracy of multiple sequence alignment, *Nucleic Acids Research* 33 pp:511-518.

[9] J. D. Thompson, F. Plewniak, and O. Poch, (1999) A comprehensive comparison of multiple sequence alignment programs, *Nucleic Acids Research* 27(13) pp:2682-2690.

[10] A. Bahr, J. D. Thompson, J. C. Thierry, and O. Poch, (2001) BAliBASE (Benchmark Alignment dataBASE) enhancements for repeats, transmembrane sequences, and circular permutations, *Nucleic Acids Research* 29(1) pp:323-326.

[11] J. D. Thompson, P. Koehl, R. Ripp, and O. Poch, (2005) BAliBASE 3.0: latest developments of the multiple sequence alignment benchmark, *Proteins* 61 pp:127-136.

[12] C. B. Do, M. S. P. Mahabhashyam, M. Brudno, and S. Batzoglou, (2005) PROBCONS: probabilistic consistency based multiple sequence alignment. *Genome Research* 15 pp:330-340.

[13] U. Roshan and D. R. Livesay, (2006) Probalign: multiple sequence alignment using partition function posterior probabilities, *Bioinformatics*, 22(22), pp:2715-21

[14] S. Miyazawa, (1995) A reliable sequence alignment method based upon probabilities of residue correspondences, *Protein Engineering* 8(10) pp:999-1009.

[15] M. O. Dayhoff, R. M. Schwartz, and B. C. Orcutt, (1978) A model for evolutionary change in proteins, In M. O. Dayhoff, editor, *Atlas of Protein Sequence and Structure,* 5 pp:345-352, National Biochemical Research Foundation, Washington DC

[16] S. Karlin and S. F. Altschul, (1990) Methods for assessing the statistical significance of molecular sequence features by using general scoring schmes, *Proceedings of National Academy of Sciences of USA,* 87(6) pp:2264-2268

[17] S. F. Altschul, (1993) A protein alignment scoring system sensitive at all evolutionary distances, *Journal of Molecular Evolution*, 36(3) pp:290-300

[18] P. H. A. Sneath and R. R. Sokal, (1973) Numerical Taxonomy. Freeman, San Francisco, CA.

[19] S. Chikkagoudar, U. Roshan, and D. R. Livesay, (2010) PLAST-ncRNA: Partition function Local Alignment Search Tool for non-coding RNA sequences, *Nucleic Acids Research*, 38, pp:W59-W63

[20] B. Paten, J. Herrero, K. Beal, and E. Birney, (2009) Sequence progressive alignment, a framework for practical large-scale probabilistic consistency alignment, *Bioinformatics* 25(3) pp:295-301

[21] U. Roshan, S. Chikkagoudar and D. R. Livesay, (2008) Searching for evolutionary distant

RNA homologs within genomic sequences using partition function posterior probabilities, *BMC Bioinformatics* 9:61

5. Figure captions

Figure 1: Hidden Markov model for pairwise sequence alignment