University of California, Berkeley Kipling Will- 23 Feb 2006

Alignment

Similarity: Two or more sequences (bases, amino acids, proteins, etc.) are matched in a **Pairwise alignment** either globally (two sequences matched over their whole length) or locally (some subset of the sequences matched while other regions are not expected to match). Sequence similarity can simply be a mathematical distance between two sequences given events such as insertions, deletions and substitutions.

In the simplest model this is the "Edit distance" or the minimal number of events required to transform one sequence into another.

Example to go from acctga to agcta:
accgta <<[substitution]>> agctga <<[deletion]>> agcta
The edit distance = 2.

BLAST (Altschul, SF, W Gish, W Miller, EW Myers, and DJ Lipman. Basic local alignment search tool. J Mol Biol 215(3):403-10, 1990).
For example, a gene is newly identified and function understood in *Drosophila*, a researcher can BLAST the database of the human genome to look for similar gene sequences.

Very basic description of BLAST
1. Uses short segments of sequence to find other sequences that contain the same set.
2. Does "ungapped" alignment extending from the matched subsequence regions to find high-scoring matches
3. Does a rapid gapped alignment to select and rank close matches

Homology: Establishing an initial estimate of homology (basically similarity) is essential. Unaligned sequence data has no *a priori* base homology. As a consequence, the fixed alignment, achieved by one method or another, is treated as prior, or background knowledge. Recall the hierarchy of characters and state and that only the states are really tested in the analyses.

The outcome of the of optimality based tree searching, especially parsimony, is strongly influenced by the alignment.

Practical issues:
Dynamic Programming and global alignment: (Needleman-Wunsch) underlies or is part of most alignment methods.
Check out tutorial at http://www.sbc.su.se/~pjk/molbioinfo2001/dynprog/dynamic.html

```
  A C T                       A G C T X
A S 1                       A 0 1 1 1 2
G 2 3                       G 1 0 1 1 2
C                           C 1 1 0 1 2
                            T 1 1 1 0 2
                            X 2 2 2 2 0
1=cost 2    A-C....
            AGC
2=cost 2    ACT
            A-G....
3=cost 1    ACT
            AGC
```

For two sequences, i.e. pairwise alignment, of length n, if no gaps are allowed then there is one optimal alignment. If gaps are allowed, i.e. there is sequence length variation, then...

$(2n)!/(n!)^2$ e.g. n=50 then 10^{29} alignments. Enumeration is not an option!

Two problems- how to find alignments and how to choose.

Taxon1 ACTTCCGAATTTGGCT
Taxon2 ACTCGATTGCCT

Minimize substitutions-

```
ACTTCCGAATTTGG-CT
|||  |||  |||  ||
ACT--CGA--TTG-CCT
```

Minimize ind/dels

```
ACTTCCGGAATTTGGCT
|||*     **|||*||
ACTC-----GATTGCCT
```

We need heuristic searches based on Optimality and scoring.

Alignment really attempts to balance the amount of indels with the amount of base substitution, normally based on some cost differential. Of course it is possible to account for all differences by inserting enough gaps (trivial alignment).

For phylogenies, pairwise comparison is not sufficient. What must be done is **multiple sequence alignment**, a global solution for the whole data matrix or primary homology for the characters (columns) in the matrix.

Various methods have been used to do this. Here are some.....

Manual or By eye- For very simple data this may be sufficient, however, it violates any criterion of repeatability as there is no obvious costs matrix. The counter argument is that the aligned matrix can be made available. However, what if I want to add or subtract OTUs? This would influence the alignment, but how? This is subject to individual pattern recognition abilities for thousands of bases and hundreds of sequences. It is also likely to increase the number of editing errors because of additional "handling" of sequences.

>>Manual alignments informed by consideration of secondary structure-

1. Does not solve the problem of nucleotide homology. At best it places constraints on changes by establishing putative limits between loop and stem regions. Nucleotides within each of those units must still be homologized and all the problems still apply.

2. Determination of secondary structure is not simple and not unambiguous. Generally the actual pattern of bonding is probabilistic and depends on the minimization of free energy and the thermodynamic stability of the resulting structure. Programs explicitly designed to model secondary structure are not very realistic (yet) in terms of the actual cell environment and might find multiple, equally probable models. In phylogenetic studies, secondary structure is typically inferred by aligning with a sequence of "known" secondary structure, although the basis of that knowledge remains uncertain and applicability to the study taxa is unclear in many cases, but this is heading in the right direction.

3. There might be reasonable to expect selective pressures to apply to secondary structure interactions (that is, requirements of compensatory changes), it is unclear just how relevant those interactions are compared to selective pressures applied at other structural levels.

>>**Purging "bad" data or scoring variable regions as single characters.**

Another method frequently used get around problems in hard to align sections is the elimination of gap heavy regions in alignments. Exactly which columns should be eliminated (left-right boundaries) is subjective and obviously they may have an impact on the results (otherwise why bother).

Alternatively, the variable region can be converted into a character in each taxon and scored. This has all the problems above and adds another layer of difficulty in determining how to code the states.

Simultaneous alignment- Simultaneous multiple alignments synchronise the information of all input sequences in a hyperspace lattice, e.g. so-called exact alignment algorithms using the divide-and-conquer (DCA) strategy (Tönges, U., Perrey, S.W., Stoye, J. and Dress, A.W.M. 1996. A general method for fast multiple sequence alignment.*Gene* **172**GC33-GC41). In part it cuts down the input sequences at carefully chosen positions to align in segments. Current algorithms cannot handle large/complex data sets.

Progressive alignment- As in Clustal W(X) the most prominent program for progressive alignment strategies.

1. All sequences are compared to each other (pairwise alignments)
2. A dendrogram is constructed, describing the approximate groupings of the sequences by similarity.
3. Final multiple alignment uses the guide tree

Basically, the multiple alignment is created by iteratively aligning sequences from the input to an already partially constructed solution. Obviously, the order is a crucial point in this method as uses a sort of UPGMA tree-based alignment order and requires sequence weighting.

It could be argued that it doesn't make sense to determine alignment order with one optimality criterion (e.g., phenetics) and then analyze the alignment later with another (e.g., parsimony, ml) but to re-align on a parsimony tree derived from the first alignment to get an "improved" alignment may be circular.

Progressive, consistency-based alignment- The genre of consistency-based multiple alignments are newer. These strategies are incorporating "local signals" into global alignment construction. See Q-align program documentation (online) for more.

Iterative, segment-based alignment- One example is DIALIGN, which iteratively collects local similar segments, which can be merged into a common multiple alignment. Iteration continues until no more local signals can be found or until all positions are aligned. Recent benchmarks have shown that this strategy can even handle long sequences of a low overall similarity. No explicit gap cost or input trees.

Direct optimization etc.- POY (W.Wheeler)- The correspondences among homologues are determined and evaluated simultaneously with transformations.

-Single process of alignment and tree construction.
-Insertion and deletion events are counted are real events (transformations) as apposed to being implied by the pattern as in multiple sequence alignments.

Promoters say-
1. Eliminates inconsistent treatment of data between alignment and tree construction steps.
2. Alignment and thus homology can be inferred under parsimony and maximum likelihood frameworks.

The program uses several methods-

Direct optimization and iterative-pass optimization strive to construct HTU sequences such that the overall cladogram is of minimal length. This is done through modified two and three dimensional string-matching, respectively.

Fixed-states optimization and search-based optimization draw optimal HTU sequences from a pool of predetermined sequences. This can be a small or large collection of possible sequences. Dynamic programming is used to identify the best HTU sequences and determine cladogram length.

Tree-alignment methods like this are ones that simultaneously deals with base changes and insertion/deletion events on the tree, with simultaneous estimation of the parameters of change (including rates in insertions and deletions, which is what in the parsimony world is referred to as gap costs, or rate matrices for nucleotide change (e.g., TV/TS ratios)).

Setting costs and implementing models is a normal part of most of these most of these methods. Determining what are the best settings is problematic.

Usually a matrix of costs or a ratio of gap:TS:TV- Results vary and there is no clear best way to choose.

Sensitivity analysis- Use a range of costs and look for character congruence and/or topological congruence. Again when is it good enough? (Terry, M. & M. Whiting. 2005. Comparison of two alignment techniques within a single complex data set: POY versus Clustal. Cladistics. 21:272-281.)

ILD= (length of combined data set on MPT -sum of the lengths of the individual data set's MPT)/length of combined data set on MPT (Mickevich & Farris.1981. The implications of congruence in *Mendia*. Syst. Zool. 30:351-370)

RILD= (length of combined data set on MPT -sum of the lengths of the individual data set's MPT)/(maximum length of the combined data set- sum of the lengths of the individual data set's MPT)) (Wheeler &Hayashi. 1998 Phylogeny of extant chelicerate orders. Caldistics. 14:173-193)