Lecture 5 – Alignment

I. Introduction. For sequence data, the process of generating an alignment establishes positional homologies; that is, alignment provides the identification of homologous phylogenetic characters.

For nucleotide data, because there are only five (if we include gaps) possible character states for each character and the states are common across all the characters, establishing character homologies can be very challenging.

Furthermore, there is no way to apply some of the criteria that are useful in positing homologies for morphological characters (transitional forms, developmental similarity). Thus, we're forced to rely on positional similarity as determined in our alignments to provide homologous characters.

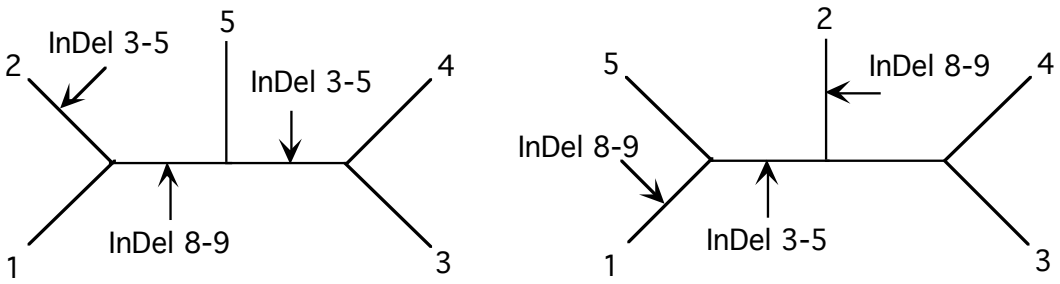
II. The ideal approach was discovered over 50 years ago. It involves **simultaneous alignment and tree estimation** (Sankoff et al., 1973). This is often called **direct optimization** (or joint estimation) and is compatible with the ideas of testing morphological hypotheses of homology with phylogenetic analyses we discussed last week.

Alignment is essentially an evolutionary inference of insertion/deletion events (indels), and is cannot be attempted logically in a non-evolutionary framework. Note that not all computer scientists adopt this view.

Let's think for a minute why alignment depends on the phylogeny.

```
1   ACCGAAT--ATTAGGCTC
2   AC---AT--AGTGGGATC
3   AA---AGGCATTAGGATC
4   GA---AGGCATTAGCATC
5   CACGAAGGCATTGGGCTC
```

So, it looks like there have be two insertion/deletion events, one at positions 3-5 and one at positions 8-9, but there's no tree on which both could have evolved only one time.



This is true for all 15 possible (unrooted) 5-taxon trees.

So really, figuring out where insertions and deletions have occurred requires a historical framework (i.e., a phylogeny), which (at least in this class) is the motivation for collecting sequence data in the first place.

We can think about alternative alignments in terms of an overall score, *D*. This includes a score for matches/mismatches and a cost for gaps: A simple score might look like this:

$$D = s + wg.$$

s is the score for a match/mismatch, *g* is a gap cost and *w* is a weighting factor for gaps.

We'll worry about the details of these values in few minutes, but the ideal goal would be to find that topology that has the best *D*, across all possible topologies and all possible alignments (i.e., find the globally optimal combination of topology + alignment; Sankoff et al. 1973).

However, given that phylogenetic estimation from a single alignment is a problem that is so computationally difficult as to require heuristic methods (short cuts), simultaneous alignment is really an incredibly difficult problem.

The most widely used heuristic in Multiple Sequence Alignment is Progressive Alignment.

III. Progressive alignment (Feng & Doolittle, 1987) is commonly used in the program **Clustal** (now in its W version).

A. Overview. Alignment occurs in three steps.

1. A **pair-wise alignment** is generated for all $(n^2\text{-}n)/2$ pairs of sequences. A **pair-wise distance** is estimated between each of the pairs and this is entered into a distance matrix.
2. That distance matrix is subjected to an algorithmic tree building procedure (usually NJ or UPGMA) to **build a guide tree**.
3. This guide tree is used to **align most closely related sequences** (which are easiest to align) with progressively more distant sequences, until all sequences are in the alignment.

B. Pair-wise alignments are conducted using Needleman-Wunsch (1970) algorithm.

Take two sequences:

```
G A A T T C A G T T A (#1)
G G A T C G A (#2)
```

So $L_1 = 11$ and $L_2 = 7$

To illustrate, we can assume a simple scoring scheme (i.e., set of alignment parameters).

$$D = s + wg$$

$S_{i,j} = 1$ if the nt at position i of sequence #1 is the same as that at position j of sequence #2

$S_{i,j} = 0$ (mismatch score)

$g = 0$ (gap penalty)

$w = 1$ (gap weight)

The idea behind Needleman-Wunsch is to generate a matrix with the two sequences to be aligned along the two axes, so in this case we have an 11 x 7 matrix. We initialize it with a row and column of zeroes (gap cost).

		G	A	A	T	T	C	A	G	T	T	A
	0	0	0	0	0	0	0	0	0	0	0	0
G	0											
G	0											
A	0											
T	0											
C	0											
G	0											
A	0											

The matrix is filled by finding the maximal score $M_{i,j}$ for each cell, where

$M_{i,j}$ = Maximum value of:

$M_{i-1, j-1} + S_{i,j}$ (match/mismatch in the diagonal), $0 + 1 = 1$

$M_{i,j-1}$ + **wg** (gap in sequence #1; vertical arrow), $0 + 0 = 0$

$M_{i-1,j}$ + **wg** (gap in sequence #2; horizontal arrow)] $0 + 0 = 0$

1 is the maximum value,

So:

		G	A	A	T	T	C	A	G	T	T	A
	0	0	0	0	0	0	0	0	0	0	0	0
G	0	1										
G	0											
A	0											
T	0											
C	0											
G	0											
A	0											

We can therefore fill in the matrix:

		G	A	A	T	T	C	A	G	T	T	A
	0	0	0	0	0	0	0	0	0	0	0	0
G	0	1	1	1	1	1	1	1	1	1	1	1
G	0	1	1	1	1	1	1	1	2	2	2	2
A	0	1	2	2	2	2	2	2	2	2	2	3
T	0	1	2	2	3	3	3	3	3	3	3	3
C	0	1	2	2	3	3	4	4	4	4	4	4
G	0	1	2	2	3	3	4	4	5	5	5	5
A	0	1	2	3	3	3	3	4	5	5	5	6

Now, we trace back from the bottom right to the top left, following the path with the highest score.

		G	A	A	T	T	C	A	G	T	T	A
	0	0	0	0	0	0	0	0	0	0	0	0
G	0	**1**	1	1	1	1	1	1	1	1	1	1
G	0	**1**	1	1	1	1	1	1	2	2	2	2
A	0	1	**2**	**2**	2	2	2	2	2	2	2	3
T	0	1	2	2	**3**	**3**	3	3	3	3	3	3
C	0	1	2	2	3	3	**4**	**4**	4	4	4	4
G	0	1	2	2	3	3	4	4	**5**	**5**	**5**	5
A	0	1	2	3	3	3	3	4	5	5	5	**6**

This gives the alignment:

```
- G A A T T C A G T T A
  |   | |   |   |     |
G G - A T - C - G - - A
```

Which has an alignment score of 6 (there are six matches). **All paths with a score of 6** represent optimal pairwise alignments, with the alignment parameters that we assumed.

So, in a multiple alignment, this is done for all $(n^2-n)/2$ pairwise comparisons of sequences.

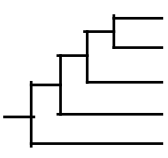
These pairwise alignments are used to estimate pairwise genetic distances, which are entered into a pairwise matrix (like the one you saw Tuesday).

C. This matrix is then subjected to an algorithmic method of generating a tree.

The Neighbor Joining method is usually used in Clustal. This is a star decomposition approach (that we'll learn soon), so it's fast.

D. This NJ tree is used to build a multiple alignment progressively, first aligning the closest sequences on the guide tree, and progressively aligning sequences that are more distant.

Guide Tree



```
1   ACCGAATATTAGGCTC
2   ACATAGTGGGATC
3   AAAGGCATTAGGATC
4   GAAGGCATTAGCATC
5   CACGAAGGCATTGGGCTC
```

Following the Guide Tree, 1 & 2 are aligned.

```
1   ACCGAATATTAGGCTC
2   AC---ATAGTGGGATC
```

Again, following the guide tree, we align sequence 3 to the fixed 1/2 alignment.

```
1   ACCGAAT--ATTAGGCTC
2   AC---AT--AGTGGGATC
3   AA---AGGCATTAGGATC
```

We continue to follow the guide tree building a multiple alignment by progressive pairwise alignments.

```
1   ACCGAAT--ATTAGGCTC
2   AC---AT--AGTGGGATC
3   AA---AGGCATTAGGATC
4   GA---AGGCATTAGCATC
5   CACGAAGGCATTGGGCTC
```

Now we have erected a series of hypothesized positional homologies and we can begin phylogeny inference.

E. Elaborations:

- A substitution matrix is used to calculate the scores for substitutions. This allows for biochemically similar amino acid substitutions to cost less than radical ones (for protein alignments) or transitional substitutions to cost less than transversional substitutions (for DNA alignments).

- Gap costs can be elaborated so that there is a separate cost associated with opening a new gap (GOP) vs, extending a gap (GEP).

- Each sequence can be weighted according to how different it is from the other sequences. This accounts for the case where one specific subfamily is over-represented in the data set.

- Position-specific gap-open penalties are modified according to residue type, using empirical observations in a set of alignments based on 3D structures. In general, hydrophobic residues have higher gap penalties than hydrophilic, since they are more likely to be in the hydrophobic core, where gaps should not occur.

- The largest portion of time is invested in conducting all the pair-wise N-W alignments to generate an initial distance matrix. There are several important shortcuts.

- MUSCLE – Uses *k*-mer words (usually $k = 6$) to derive a distance (just like the Edwards et al. method). This is much faster than doing a bunch of pair-wise NW alignments, and works just as well. As you might expect, the more similar sequences are, the more hexamers they'll have in common.

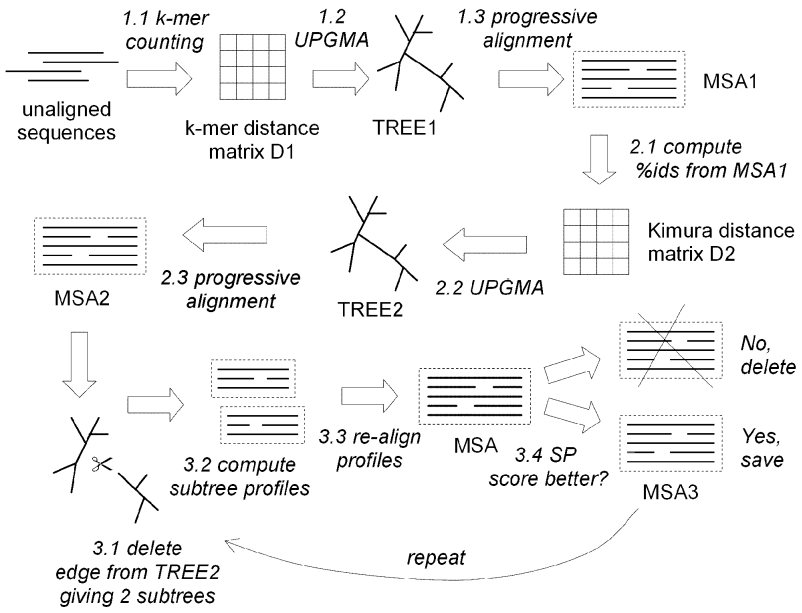


Figure 2. This diagram summarizes the flow of the MUSCLE algorithm. There are three main stages: Stage 1 (draft progressive), Stage 2 (improved progressive) and Stage 3 (refinement). A multiple alignment is available at the completion of each stage, at which point the algorithm may terminate.

F. Models in Alignment

A big issue is that the final alignment is dependent on the parameters, and those parameter values depend on an unknown processes of evolution (i.e., insertion rate, deletion rate, and rates of various substitution types).

The Catch-22 is that we need a good tree to estimate these. This is because, as discussed in the beginning of this lecture, alignment and historical information are not independent of each other.

A growing body of work attempts to model this so that we can simultaneously optimize parameters and alignments.

These are extremely computationally intensive.

There are two approaches:

1. SATé (& SATé II: Liu et al. 2012. Syst. Biol. 61:90) is an iterative (successive approximations) approach that bundles MAFFT, MUSCLE & RAxML (which we'll discuss later).

   It uses models of nucleotide substitution, including rate variation among sites (which we'll also talk about) and ML estimation of intermediate trees.

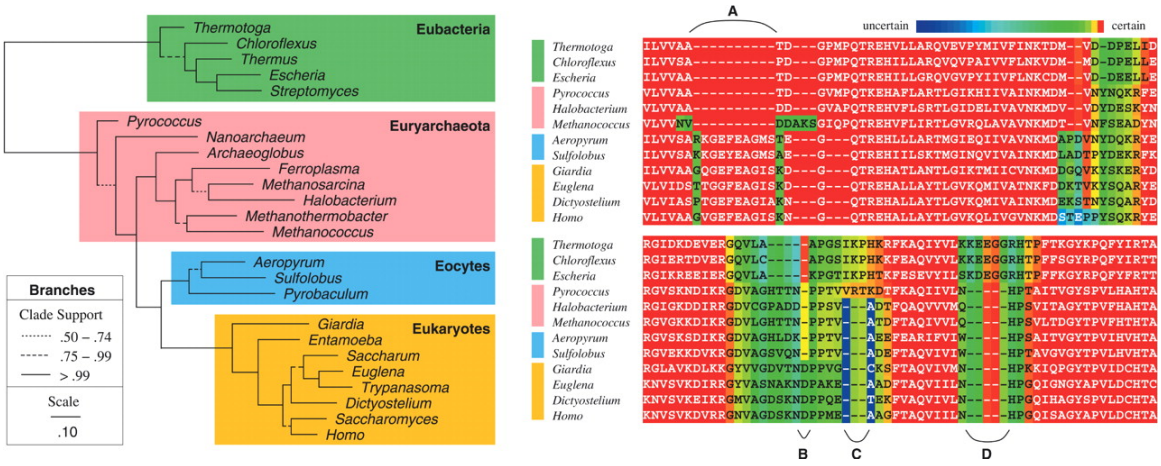
   However, its improvements in alignment are mostly derived by the subtree pruning (like we just saw); this is pretty extensive and permits a maximum of 32 taxa per subtree.

2. Fully parametric Bayesian approaches have been developed (Redelings and Suchard 2005; Suchard and Redelings 2006)) that conduct simultaneous alignment and phylogeny estimation, and these have the advantage that they integrate across uncertainty in alignments.

   explores a vast state space: $\boldsymbol{\omega} = (\mathbf{Y},\mathbf{A},\tau,\mathbf{T},\boldsymbol{\Theta},\boldsymbol{\Lambda})$.

   Y = Unaligned sequences
   A = Alignment
   τ = The tree
   T = Its branch lengths
   Θ = The substitution model
   Λ = The model of insertion/deletion

At this point, though, they're only applicable to pretty moderate data sets.



It actually provides an explicit evaluation of the uncertainty in alignment sites.

A really good recent paper, Wygoda et al. (2024) explores Λ, the model of InDel evolution and provides a means of selecting among these modesl.

A cool workaround to the computational demands of a fully Bayesian approach to accounting for uncertainty in alignment is Ashkenazy et al. (2019), where averaging across alignments is accomplished by concatenating alternative alignments into a Super MSA.

A very abbreviated list of relevant references.

Ashkenazy, H., I. Sela, E. L. Karin, G. Landan, & T. Pupko. 2019. Multiple sequence alignment averaging improves phylogeny reconstruction. Syst. Biol., 68:117-130.

Edgar, R. C. 2004. Local homology recognition and distance measures in linear time using compressed amino acid alphabets. Nucleic Acids Res., 32:380-385.

Edgar, R. C. 2004. MUSCLE: multiple sequence alignment with high accuracy and high throughput. 32:1792-1797.

Feng, D.-F. and R. F. Doolittle. 1987. Progressive sequence alignment as a prerequisite to correct phylogenetic trees. J. Mol. Evol. 25: 351-360.

Hein, J. J., L. Jensen, & C. N. S. Pedersen. 2003. Recursions for statistical multiple alignment. PNAS 100:14950-14965.

Liu, K. T. J. Warnow, M. T. Holder, S. M. Nelesen, J. Yu, A. P. Stamatakis & C. R. Linder. 2012. SATé II: Very fast and accurate simultaneous multiple sequence alignments and phylogenetic trees. Syst. Biol. 61:90-106.

Metzler, D., R. Fleißner, A. Wakolbinger, & A. von Haeseler. 2001. Assessing variability by joint sampling of alignments and mutation rates. J. Mol. Evol. 53:660-669.

Needleman, S. B. & C. D. Wunsch. 1970. A general method applicable to the search for similarities in the amino acids sequence of two proteins. J. Mol. Biol. 48:443-453.

Redelings, B. D. and M. A. Suchard. 2005. Joint Bayesian estimation of alignment and phylogeny. Syst. Biol., 54:401-418.

Redelings, B.D. 2021. BAli-Phy version 3: Model-based co-estimation of alignment and phylogeny. 37:3020-3024. https://academic.oup.com/bioinformatics/article/37/18/3032/6156619

Sankoff D., C. Morel, & R. J Credergren. 1973. Evolution of 5S RNA and the non-randomness of base replacement. *Nature New Biology*, 245:232-234.

Smith, T. F. & M. S. Waterman. 1982. Identification of common molecular subsequences. J. Mol. Biol. 147:195-197.

Suchard, M.A and B. D. Redelings. 2006. BAli-Phy: simultaneous Bayesian inference of alignment and phylogeny. Bioinformatics, 22:2047-2048.

Thompson, J. D., D. G Higgins, and T. J. Gibson. 1994. CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position specific gap penalties and weight matrix choice. Nucleic Acids Res. 22:4673-80.

Wheeler, W. 2001. Homology and the optimization of DNA sequence data. Cladistics,17:S3-S11.

Wygoda, E., G. Loewenthal, A. Moshe, M. Albuquerque, I. Mayrose, & T. Pupko. 2024. Statistical framework to determine indel-length distribution. Bioinformatics, 40: https://doi.org/10.1093/bioinformatics/btae043