

HMMs III: Weighting Data.

Multiple Sequence Alignment: Alignment Unknown.

In most applications, you will not be given the alignment of sequences, but will have to derive this. This is very difficult, in general, because of dimensional/complexity reasons: if we do pairwise sequence alignments by dynamic programming, we will need to compute the entries of an $M \times N$ matrix, where M, N are the lengths of the two sequences. If we have k sequences of length M_1, M_2, \dots, M_k , then the corresponding DP algorithm would require $O(M_1 \cdot \dots \cdot M_k)$ operations. For $k = 3$ this is feasible, if slow. For k larger it still seems to be prohibitive. Notice that if we have k sequences to align, even the number of gap patterns we have to check at each stage of the computation grows exponentially with k !

There are several issues here: even if we could do DP for an array of sequences, how would we score it? If we want to align k sequences, do we

need to have curated sequences reliably aligned so that we can perform counts to get the probabilities $p(a_1, \dots, a_k)$ of k specific residues being present in a column? Notice we have even less idea than before which way the “direction” of evolution or substitution went in this case).

Scoring MSAs.

Because of lack of data, one uses a variety of substitutes for complete counts.

1.) Sum of Pairs: Here one simply gives the score of a column as $\sum_{i,j} s(a_i, a_j)$, where a_i, a_j run over the residues observed in the alignment. (Usually a gap is just treated as another residue.) The total score is just adding the various column scores up (so that we are, in effect, assuming the columns are statistically independent.)

2.) Entropy measures: Here one assumes the positions within the columns are independent and uses a simple entropy measure. If $p_{a,i}$ is the probability of residue a in the column i , then the measure is

$$-\sum c_{a,i} \log p_{a,i},$$

which is, up to a scale factor, the entropy of the column distribution. Complete conservation would give 0 as the entropy (and only that!), and one would try to minimize this kind of score.