



Sequence Matching: The Needleman Wunsch Algorithm.

Constructing an alignment:

- The alignment score is cumulative by adding along a path through the array.
- The best alignment is the highest score, i.e., the maximum match.
- The maximum match will always be somewhere in the outer row or column.
- The alignment is constructed by working backwards from the maximum match.



Tools for Alignment and Matching:

We have seen the Smith-Waterman and the Needleman-Wunsch algorithms for matching sequences.

Using these algorithms, given two sequences of length  $m$  and  $n$ , we must compute a table of size  $n \times m$ . In each algorithm, each entry in this table takes a constant amount of computation. If this constant is  $t_c$ , then the time taken for one match is

$$T = n \times m \times t_c$$

When  $n$  and  $m$  become large (even for simple proteins,  $n$  and  $m$  can be tens of thousands), these algorithms can take a long time to execute.

Furthermore, we are typically matching a reference sequence against a large database of sequences. Therefore, we might need to perform a large number of such matches. In this case, approximations to these algorithms might be necessary.

Sequence Matching: The Needleman Wunsch Algorithm.

Just as in the Smith-Waterman algorithm, we can augment this algorithm with similarity tables and gap penalties.

In our example, we had used a trivial similarity measure: match = 1, no match = 0.

We had also used gap penalty = 0.

In this case,

$$H_{ij} = \max\{ \begin{aligned} &H_{i-1,j-1} + s(a_i, b_j), \\ &\max_k\{H_{i-k,j-1} - W_k + s(a_i, b_j)\}, \\ &\max_r\{H_{i-1,j-r} - W_r + s(a_i, b_j)\} \end{aligned} \}$$