Dynamic Prog Algo

$$\begin{array}{ll} & s_1\ s_2\ s_3\ s_4 \\ S: & A\ C\ A\ T \\ T: & A\ G\ T \\ & t_1\ t_2\ t_3 \end{array}$$

$$X = \begin{array}{c|cccc} & \cdot & A & G & T \\ \hline \cdot & & & & \\ A & & & & \\ C & & & & \\ A & & & & \\ T & & & & \bigcirc \end{array}$$

$X_{i,j}$ = Score of the best alignment for $s_1 \ldots s_i$ against $t_1 \ldots t_j$

Note: We want $X_{m,n}$, and we want the alignment that achieves this score.

How to calculate $X_{i,j}$?

Best Aln $\begin{pmatrix} s_1 \ldots s_i \\ t_1 \ldots t_j \end{pmatrix}$ either looks like

Alignment	Score
Best Aln $\begin{pmatrix} s_1 \ldots s_{i-1} \\ t_1 \ldots t_{j-1} \end{pmatrix} \begin{matrix} s_i \\ t_j \end{matrix}$	$\rightarrow X_{i-1,j-1} + M($
OR	
Best Aln $\begin{pmatrix} s_1 \ldots s_{i-1} \\ t_1 \ldots t_j \end{pmatrix} \begin{matrix} s_i \\ - \end{matrix}$	$\rightarrow X_{i-1,j} + c$
OR	
Best Aln $\begin{pmatrix} s_1 \ldots s_i \\ t_1 \ldots t_{j-1} \end{pmatrix} \begin{matrix} - \\ t_j \end{matrix}$	$\rightarrow X_{i,j-1} + c$

So: $X_{i,j} = \max \begin{cases} X_{i-1,j-1} + M(s_i, t_j) \\ X_{i-1,j} + c \\ X_{i,j-1} + c \end{cases}$ (when $i>0, j>0$)

$X_{0,j} = j \cdot c$
$X_{i,0} = i \cdot c$
$X_{0,0} = 0$

M: [ ]

c = -2

	.	A	G	T
.	0	-2	-4	-6
A	-2	↖ +1	← 0	← -2
C	-4	↑ -1	↖ ~~0~~ 1	-1
A	-6	↖ ↑ -3	↖ -2	↖ ↑ -3
T	-8	↑ -5	↑ -4	↖ -1

Calculate X entri
to right, top to

For each entry, kee
to entry that achieved

$X_{4,3} = -1$ = Score of best alignment

We recover best alignment following backpointers from (m,n)

↖ : Match

← : Gap in S

↑ : Gap in T

Best	= A C A T
Aln	A - G T

Running time: O(m·n)

Space: O(m·n)

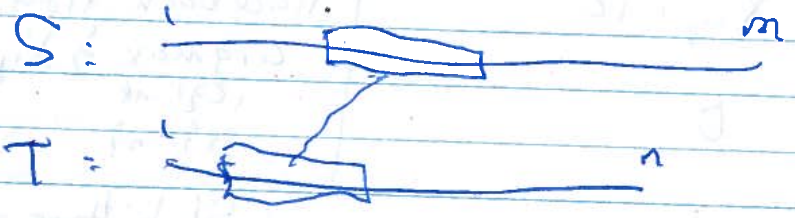
Note: NW algo can be modif
with affine gap pena
→ Why does it not al
→ What is the modific

To recover all optimal alignments, do Depht-First-Search e
backpointers starting from (m,n).

Pairwise local alignment problem

Idea: Often, two sequences only share significant sim
over a particular region.
Examples: Two

⋮



Given: { Sequences $S = s_1 \ldots s_m$ and $T = t_1 \ldots t_n$
Subst. matrix M
gap penalty c

Find: ~~[illegible]~~ Indices i, j, k, ℓ, where $\begin{cases} 1 \leq i \leq j \leq m \\ 1 \leq k \leq \ell \leq n \end{cases}$

such that $\text{Score}\left(\text{Best Aln}\begin{pmatrix} s_i \ldots s_j \\ t_k \ldots t_\ell \end{pmatrix}\right)$ is maximized

Can we do better than:

```
for i = 1...m
    for j = i...m
        for k = 1...n
            for ℓ = k...n
                NW(s_i...s_j, t_k...t_ℓ)
```

$\Big\} O(m^3 n^3)$

Smith Waterman algo.

$X_{i,j}$ = Score of best local alignment for $s_1 \ldots s_i$, $t_1 \ldots t_j$ where s_i and t_j are included in alignment

$$X_{i,j} = \max \begin{cases} X_{i-1,j-1} + M(s_i, t_j) \\ X_{i-1,j} + c \\ X_{i,j-1} + c \\ 0 \end{cases}$$

$X_{i,0} = 0 \quad \forall i = 1 \ldots m$
$X_{0,j} = 0 \quad \forall j = 1 \ldots n$

Traceback from ~~[illegible]~~ $\underset{\substack{i \in \{1 \ldots m\} \\ j \in \{1 \ldots n\}}}{\operatorname{argmax}} \{X_{i,j}\}$

until hitting zero