LOCAL ALIGNMENT

X = ACGTTA ; (positions 1 2 3 4 5 6)

how many substrings in string of length n?

$\frac{1}{2}n^2 - \frac{n}{2} \longrightarrow O(n^2)$

prefixes (X) = {A, AC, ACG, ACGT, ACGTT, ACGTTA}

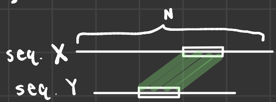
suffixes (X) = {A, TA, TTA, GTTA, CGTTA, ACGTTA}

substrings (X) = { A, AC, ACG, ACGT, ACGTT, ACGTTA,
C, CG, CGT, CGTT, CGTTA,
G, GT, GTT, GTTA,
T, TT, TTA,
T, TA,
A }

prefixes (suffixes (X))
suffixes (prefixes (X)) } both give all substrings of X!

this concept is fundamental to the local alignment algorithm

Local alignment:



size of optimal local alignment = O(log N)

<u>intuition:</u> look @ every subsequence from X and Y sequences ⟶

look for subsequences with highest global alignment score! yay!

but we have quadratically many subseq. of X and Y ... uh oh this could be computationally very inefficient → enter SMITH-WATERMAN

<u>The Local Alignment Problem:</u>

- <u>given:</u> 2 sequences X and Y and scoring scheme δ
- <u>find:</u> 2 subsequences α and β of X and Y whose optimal global alignment is <u>MAXIMAL</u> over all pairs of subsequences from X and Y.
  - Brute force comparison of substrings : $O(m^2 n^2)$ — len(X) = m
  - However, Smith-Waterman : $O(mn)$ ~ quadratic — len(Y) = n

▷ $v(i,j)$ = value of optimal global alignment btwn suffixes of X_i (prefix length i of X) and Y_j (prefix length j of Y)

▷ v^* = OPTIMAL LOCAL ALIGNMENT

$$\max [v(i,j)), \; 0 \le i \le m, \; 0 \le j \le n]$$

<u>proof</u> : (you won't be tested on this, but it's good to think about :))

1. $v^* \ge \max[v(i,j), \; 0 \le i \le n, \; 0 \le j \le m]$ ~ trivial
2. want to show $v^* \le \max[v(i,j), \; 0 \le i \le n, \; 0 \le j \le m]$ because then we can say they're equal

- v^* = optimal solution to our problem ~ say α and β = optimal local alignment

→ $v^* = v(i^*, j^*) \le \max[v(i,j), 0 \le i \le n, 0 \le j \le m]$

⟶ since we proved $v^* \ge \max[v(i,j), \; 0 \le i \le n, \; 0 \le j \le m]$ AND

$v^* \le \max[v(i,j), \; 0 \le i \le n, \; 0 \le j \le m]$, we know

$$v^* = \max[v(i,j), \; 0 \le i \le n, \; 0 \le j \le m]$$

The SMITH-WATERMAN ALGORITHM:

$V(i,0) = 0, \ \forall i, \ 0 \le i \le m$

$V(0,j) = 0, \ \forall j, \ 0 \le j \le n$

For $i > 0$ and $j > 0$:

$$V(i,j) = \max \begin{cases} 0, \\ V(i-1,j) + \delta(x_i, -) \\ V(i, j-1) + \delta(-, y_j) \\ V(i-1, j-1) + \delta(x_i, y_j) \end{cases}$$

More details to come ... 🐮