

Ali Ramazani

Question 1a:

Algorithm Idea: The algorithm will fill in a table to compute the scores of all possible alignments of substrings of v and w . The key idea is to combine local alignments of v with global alignments of w to find the optimal fitting alignment.

Initialization:

T is a table of size $(m+1) \times (n+1)$ with zeros.

$T[i, j]$ gives us the score of the fitting alignment of the substrings $v[1 \dots i]$ and $w[1 \dots j]$.

$T(i, 0) = 0$ for $0 \leq i \leq m$

$T(0, j) = 0$ for $0 \leq j \leq n$

Recurrence:

$$T[i, j] = \max \begin{cases} T[i-1, j-1] + \alpha(v_i, w_j) & \text{match/mismatch} \\ T[i, j-1] + \sigma & \text{gap in } v \\ T[i-1, j] + \sigma & \text{gap in } w \\ 0 & \text{start a new alignment} \end{cases}$$

Return:

The optimal score for the fitting alignment is stored in the last row of Table T

$\max\{T(m, j)\}$ for $0 \leq j \leq n$

Here, m is the length of string v , and n is the length of string w .

Runtime: $O(nm)$

1b) For a single 250bp read, fitting alignment might be a reasonable approach. Considering the run-time of our fitting algorithm, $O(250 * 3 \text{ billion}) = O(750 \text{ billion})$ base pairs, which would take about 1.83 GB of storage space, assuming 3 billion bp takes about 725 MB.

1c) Considering $O(250 \text{ million} * 3 \text{ billion}) = O(750 \text{ trillion})$ base pairs, which takes about 1826 GB of storage. Therefore, fitting alignment would not make sense here because aligning a million short reads to a larger genome using dynamic programming requires significant memory, which can be a constraint on a typical desktop computer.

Note: I used this [stackoverflow](#) to help me answer parts b and c. I also talked with Carlos and shared ideas to help each other.

Data Analysis:

```
python3 local_alignment.py TP53-Human.fasta TP53-Cat.fasta scoring1.txt
```

Alignment Score: 28.0

Optimal Local Alignment:

```
GCACATCTGCAT-TTTCACCC-CACCCTTCCCCTCCTTCTCCC-TTTTTATATCCCATTTTATA CG
GCACATCTGCGTATTTT-CCCACACCCTTCCCC-C-T-CTCCCCTTTTATATCCCCTTTTATATCG
```

```
python3 local_alignment.py TP53-Rat.fasta TP53-Cat.fasta scoring1.txt
```

Alignment Score: 8.0

Optimal Local Alignment:

```
GACTCTGT
```

```
GACTCTGT
```

```
python3 local_alignment.py TP53-Rat.fasta TP53-Human.fasta scoring1.txt
```

Alignment Score: 8.0

Optimal Local Alignment:

```
GGCCAGCC
```

```
GGCCAGCC
```

Based on the alignment scores, the Human and Cat sequences appear to be the most similar in the aligned regions.