Needleman-Wunsch Algorithm

Hassan Jaber

Contents

Report

I. Problem Description:

In this project we are going to implement a specific algorithm which is called ”Needleman-Wunsch”.

This algorithm is used to find the global sequence alignment (not the local alignment) between two DNA sequences or between two protein sequences.

After implement it we need to test it:

- Firstly we want to test it using two homologous genes that are differing around 10% and the testing should be using two different scoring functions(match, mismatch, gap).

For these two homologous I chose a gene called ”lysine acetyltransferase 5 (KAT5)” also known as “TIP60” from Humans(Homo Sapiens)(ref:1) and Sperm Whales(Physeter catodon)(ref:2) because this exact gene is quite similar between these two species, and it’s an interesting gene.

What is ”lysine acetyltransferase 5 (KAT5)” exactly?

The protein encoded by this gene belongs to the MYST family of histone acetyl transferases (HATs) and was originally isolated as an HIV-1 TAT-interactive protein. HATs play important roles in regulating chromatin remodeling, transcription and other nuclear processes by acetylating histone and nonhistone proteins. This protein is a histone acetylase that has a role in DNA repair and apoptosis and is thought to play an important role in signal transduction. Alternative splicing of this gene results in multiple transcript variants. (ref: 3).

- Secondly we are going to do the same thing as above but with other two sequences, Human Insulin Protein and Hamster Insulin Protein.

II. Methods:

The methods used in this project are:

- Python 3.10 for dynamic programming.

- Numpy library: It’s mainly used for working with arrays and matrices (for example: initializing matrices, formatting and reshaping matrices…).

- IPython library: for having a nicely displayed output in the console.

- Jupyter Notebook as an IDE to run the code.

- Needleman-Wunsch Algorithm.

- https://www.ncbi.nlm.nih.gov/ Website for downloading the FASTA format files used for comparison, and for having a general information about genes and proteins.

- https://bioboot.github.io/bimm143_W20/class-material/nw/ Website to check if my implementation of the algorithm was correct.

- Two scoring function used in comparisons:
  match = 4, mismatch = -2, gap = 0
  and
  match = 2, mismatch = -1, gap = -2.

III. Results

First of all, when I compared the two results from the comparison of the homologous genes, I found out that when I had gap score = 0 the output would have an optimal alignment because every two character from the two sequences are aligned with each other perfectly, we can check that in the output.txt or in the output console when we run the algorithm class, but when I had gap score = -2 the output alignment was not optimal for some reason.

Secondly, when I did the same comparison for the human and hamster insulin I got the same result.
When gap = 0 the output was perfect optimal alignment but when gap = -2, the alignment was not optimal.

Thirdly, when match score is higher we get an higher score from the algorithm regardless of the output is optimal or not.

Fourthly, changing the mismatch did not affect the alignment it only affected the score matrix, as long as mismatch < 0 or we would get a completely wrong alignment (not even close to the optimal one).

IV. Discussion

In conclusion, personally, I think in order to always have the best optimal alignment between two sequences we have to set the gap = 0 and mismatch <0 and match >0.

Remarks: 1) When I try to output a very big matrix in jupyter notebook I would always get an error if I do not use this commend in the anaconda terminal: “jupyter notebook --NotebookApp.iopub_data_rate_limit=1.0e10” because the output console in jupyter notebook is limited by default so there would not be a crash, so we have to tweak it by the above commend.

2) Upon opening jupyter notebook please run the main code just once so you can see the nicely formatted matrices clearly in the output.

3)Every time you initiate the algorithm Class, a new optimal alignment will be overwritten in the output.txt file.

V. References

Ref 1: https://www.ncbi.nlm.nih.gov/nuccore/NM_182710.3?report=fasta

Ref 2: https://www.ncbi.nlm.nih.gov/nuccore/XM_007128188.3?report=fasta

Ref 3: https://www.ncbi.nlm.nih.gov/gene/10524