Assignment – III

(Sequence Alignment)

Deadline: 10th April

1. You are given 2 nucleotide sequences:

GGCTGCAACTAGCTC

GGGTAAGCTTGC

and the transition-transversion scoring matrix (expressed in similarity):

	A	C	G	T
A	4	-1	1	-1
C	-1	4	-1	1
G	1	-1	4	-1
T	-1	1	-1	4

and gap penalty -3.

Carry out the global and local alignment (dynamic programming algorithm), and indicate the final similarity score and the best alignment. (done)

2. Identify the dinucleotide CA repeat region and the score in the following sequence:

TGGCACACTCACACCACACAGACAGTTA

3. When would you encounter a situation for using DP for overlap regions? How are the boundary conditions and recursive relations different from that for global alignment?

(done)

4. What is the advantage of using affine gap scores? (done)

5. Give the time and space complexity of DP. Under what conditions is time an issue and under what conditions would space be a problem? (done)

6. Describe the construction of Nucleic acid PAM scoring matrices. (done)

7. Take any gene sequence and its corresponding protein sequence and perform databases searches with both these sequences. Which of these two searches identifies more significant matches? Give reasons. (done)

8. What is the difference in the working of PSI-BLAST and BLAST programs? (done)

9. (i) In BLAST database search algorithm, the match/mismatch ratio for comparing nucleotide sequences is chosen to be large for highly conserved sequences, while it is small for divergent sequences. Give reasons, why?

(ii) Give the BLAST nucleotide substitution matrix for comparing sequences that are 95% conserved.

(not sure what the matrix is : have added the info given in the slides)

10. In the BLOSUM62 matrix, a conserved Tryptophan position has score S(W,W) = 11, but a conserved Leucine position has score S(L,L) = 4. Give at least one reason why these values differ. (done)

11. Construct the scoring scheme for identifying DNA sequences that exhibit at least 65% identity. Assume background frequency 0.25, for each of nucleotides and assume equiprobability for mismatch. (not done at all)

SOLUTIONS :

1. The solutions are as follows:

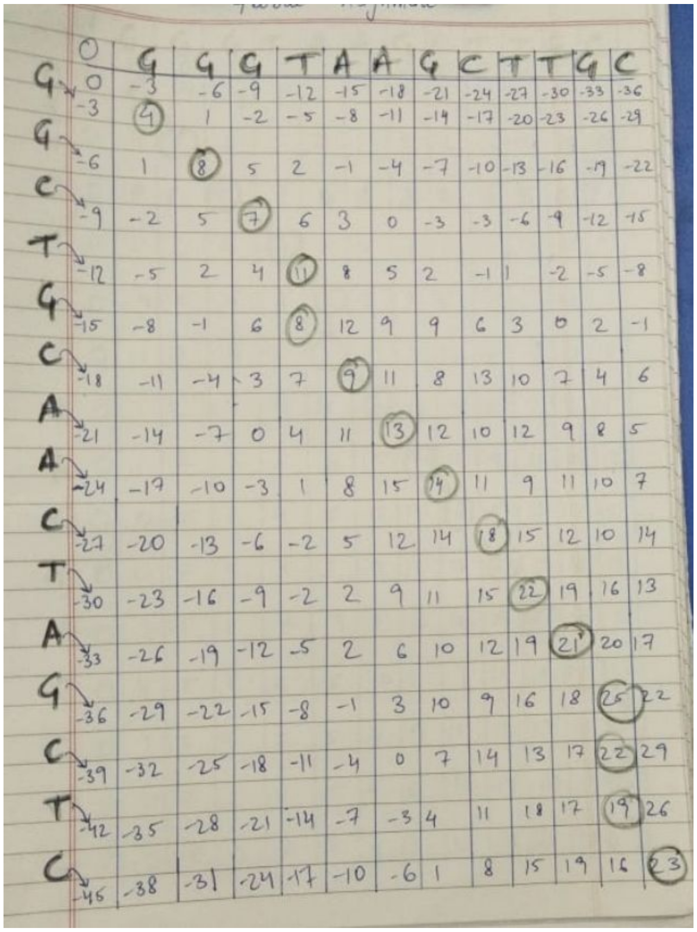
a. Global Alignment :

i. Initialise F(0,0) = 0

ii. Boundary conditions : F(i.0) = F(0,i) = -i*d

iii. F(i,j) = max { F(i-1 , j-1) + s(xi, yj) , F(i-1, j) - d, F(i, j-1)-d }

iv. The matrix is as follows:



	0	G	G	G	T	A	A	G	C	T	T	G	C
	0	-3	-6	-9	-12	-15	-18	-21	-24	-27	-30	-33	-36
G	-3	4	1	-2	-5	-8	-11	-14	-17	-20	-23	-26	-29
G	-6	1	8	5	2	-1	-4	-7	-10	-13	-16	-19	-22
C	-9	-2	5	7	6	3	0	-3	-3	-6	-9	-12	-15
T	-12	-5	2	4	11	8	5	2	-1	1	-2	-5	-8
G	-15	-8	-1	6	8	12	9	9	6	3	0	2	-1
C	-18	-11	-4	3	7	9	11	8	13	10	7	4	6
A	-21	-14	-7	0	4	11	13	12	10	12	9	8	5
A	-24	-17	-10	-3	1	8	15	14	11	9	11	10	7
C	-27	-20	-13	-6	-2	5	12	14	18	15	12	10	14
T	-30	-23	-16	-9	-2	2	9	11	15	22	19	16	13
A	-33	-26	-19	-12	-5	2	6	10	12	19	21	20	17
G	-36	-29	-22	-15	-8	-1	3	10	9	16	18	25	22
C	-39	-32	-25	-18	-11	-4	0	7	14	13	17	22	29
T	-42	-35	-28	-21	-14	-7	-3	4	11	18	17	19	26
C	-45	-38	-31	-24	-17	-10	-6	1	8	15	19	16	23

v. Final similarity score = 23

vi. Best Alignment:

1. GGCTGCAACTAGCTC
2. _GGGTAAGCTTG__C

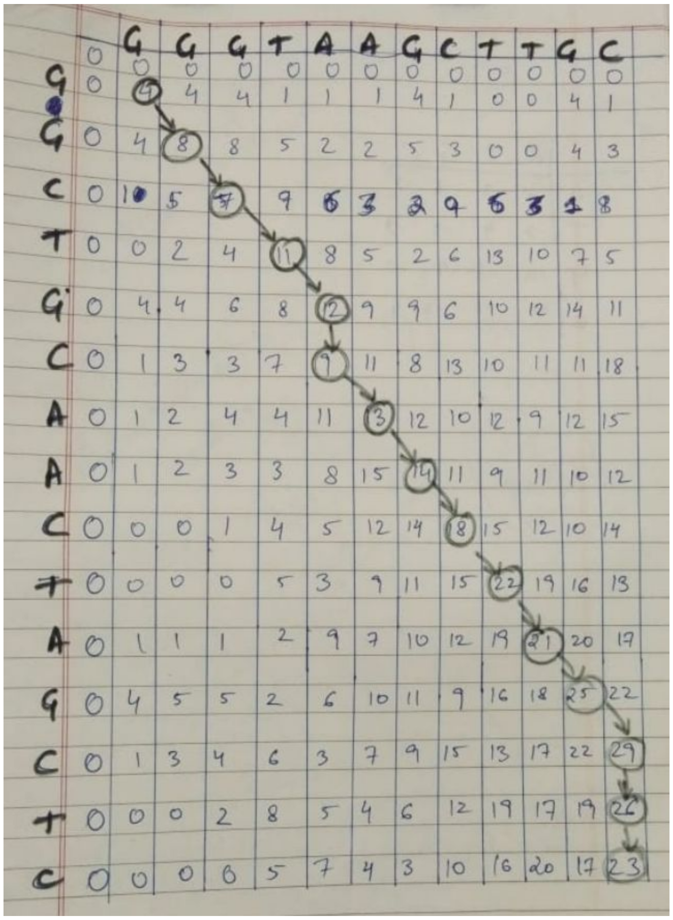
b. Local Alignment

i. Initialize: F(0, 0)=0

ii. Boundary conditions: F(i, 0) = F(0, i) = - id

iii. F(i, j) = max{0, F(i-1, j-1) + s(xi , yj), F(i-1, j) - d , F(i, j-1) – d}

iv. The matrix is as follows:



		G	G	G	T	A	A	G	C	T	T	G	C
	0	0	0	0	0	0	0	0	0	0	0	0	0
G	0	4	4	4	1	1	1	4	1	0	0	4	1
G	0	4	8	8	5	2	2	5	3	0	0	4	3
C	0	10	5	7	9	6	3	2	9	5	3	1	8
T	0	0	2	4	11	8	5	2	6	13	10	7	5
G	0	4	4	6	8	12	9	9	6	10	12	14	11
C	0	1	3	3	7	9	11	8	13	10	11	11	18
A	0	1	2	4	4	11	13	12	10	12	9	12	15
A	0	1	2	3	3	8	15	14	11	9	11	10	12
C	0	0	0	1	4	5	12	14	18	15	12	10	14
T	0	0	0	0	5	3	9	11	15	22	19	16	13
A	0	1	1	1	2	9	7	10	12	19	21	20	17
G	0	4	5	5	2	6	10	11	9	16	18	25	22
C	0	1	3	4	6	3	7	9	15	13	17	22	29
T	0	0	0	2	8	5	4	6	12	19	17	19	26
C	0	0	0	6	5	7	4	3	10	16	20	17	23

v. Final Similarity Score = 23

vi. Best Alignment:

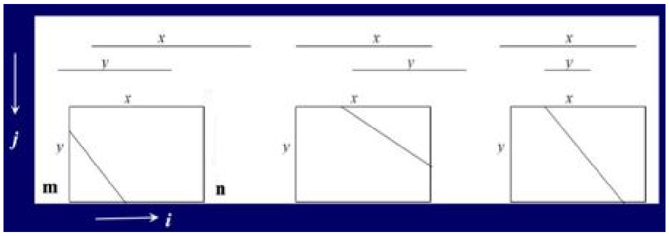
1. GGCTGCAACTAGC
2. GGGT_AAGCTTGC

2. Solution is as follows:

a. The sequence is TGG CACAC TCACAC CACACAGACAGTTA
b. Here , the Tandem repeat region is CACAC
c. Score for the same is 20

3. Overlap sequences occur when one sequence is contained in the other, or they have common overlapping regions. e.g., when comparing fragments of genomic DNA sequence to each other, or to large chromosomal sequences, in sequence assembly.

a. Now for two sequences of length n , different types of alignments are possible.

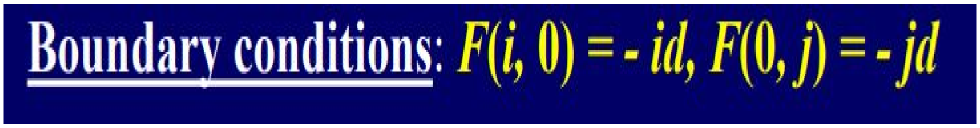
i. 

ii. For every possible type combined, there can be a huge number of alignments in total:

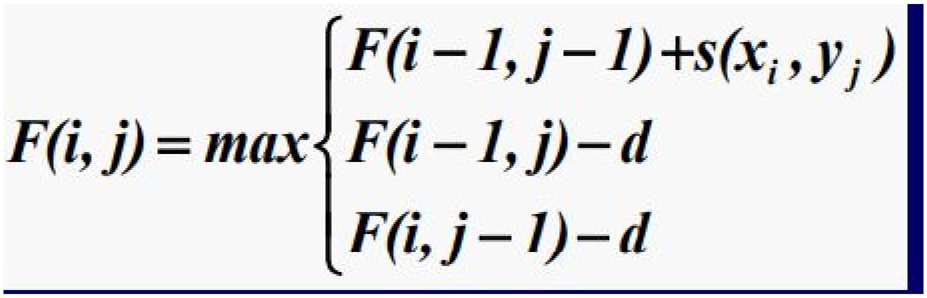
1.
2. Dealing with all of them is not computationally feasible. Hence we use Dynamic Programming.

b. The boundary conditions and recursive relations for global and local alignment are as follows :

   i. Global Alignment :



Boundary conditions: $F(i, 0) = - id, F(0, j) = - jd$



$$F(i, j) = max \begin{cases} F(i-1, j-1) + s(x_i, y_j) \\ F(i-1, j) - d \\ F(i, j-1) - d \end{cases}$$

   ii. Local Alignment :

$$F(i, 0) = 0, F(0, j) = 0$$

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

4. Affine Gap Scores :

a. Affine Gap Scores are given by Cost(g) = - d – (g-1)e [g = gap length, d = gap-open penalty, e = gap-extension penalty] While using this gap cost, we need to keep track of multiple values for each pair of i,j .

b. Whilst using this method, there is an underlying assumption that a deletion will not be followed directly by an insertion (true for the optimal path if – d – e is less than the lowest mismatch score) . Another way of looking at this is that, an affine score assumes that consecutive deletions/insertions are a single mutation event as opposed to multiple insertions/deletions and hence, should be penalised less.

c. The advantage behind the use of Affine Score Gaps is that Affine gap versions provide the most sensitive sequence matching methods.

5. The space complexity and time complexity of DP are both O(n*m). With these orders of magnitudes , there are some constraints on the time and size usage . They are as follows :

a. Dynamic Programming implementation is not feasible for comparing complete genomes of chromosomes (i.e a sequence that is more than a few Mbs long) . In such cases, space becomes an issue and the space complexity , thus, needs to be addressed.

b. Whilst using DP, If in a database search, a sequence of length n is searched in a database sizing around a few Gbs , the time complexity can become an issue and will thus need to be addressed.

6. PAM stands for Point Accepted Mutations (or Percent of Accepted Mutations) . They are derived from global alignments of closely related sequences , and these matrices refer to various degrees of sensitivity depending on the evolutionary distance between sequence pairs. The construction details are as follows :

a. The construction is based on the hypothesis that proteins diverge by accumulating uncorrelated mutations.

b. We align closely related sequences (> 85 % identity) considering that very small sequences allow the correct alignments to be determined with high certainty.

c. We observe the probability of AA changes and compute the log-odds ratio.

d. We then normalise the matrix (relative frequencies of various mutations multiplied by a carefully chosen constant) to give an average change of 1% of all positions to obtain the PAM-1 matrix

e. Now that we have the PAM-1, should we want to derive matrices for distantly related sequences from data about closely related sequences, we'll do so by extrapolating from PAM-1 by successive iteration of the reference mutation matrix:

   i. $M_n = (M_1)^n$

f. Assumption in this evolutionary model is that AA substitutions over short periods of evolutionary history can be extrapolated to longer distances.

7. The search results and their reason are as follows (The protein sequence and DNA are that of SARS-COV2):

a. Protein Based Search:

   i. https://www.ebi.ac.uk/Tools/services/web/toolresult.ebi?jobId=fasta-I20200404-183206-0162-65266355-p2m

b. Dna Based Search:

   i. https://www.ebi.ac.uk/Tools/services/web/toolresult.ebi?jobId=fasta-I20200404-183343-0848-74276617-p2m

c. As we can see from the results, we get more significant matches for the protein search. The reasons are as follows

   i. There are very different DNAs that code for similar protein sequences, hence we'll get a higher number of matches here.

   ii. When comparing DNA sequences, we get significantly more random matches than we get with proteins. The reason for the same is that:

      1. DNA is made of just 4 different characters (A , T , G , C). Thus , even two unrelated DNA are expected to have approximately 25% similarity.
      2. In contrast, a protein sequence is composed of around 20 different Amino Acids. This definitely improves the sensitivity of the comparison. Thus matches happening with proteins usually come from homologues.
      3. DNA databases are much larger, and grow faster than protein databases, and thus experience more random hits.

   iii. For DNAs, we usually use identity matrices, while for proteins we use more sensitive matrices like PAM and BLOSUM. These usually result in better search results.

iv. Proteins are rarely mutated during evolution. Due to their conservation, searching them reveals remote evolutionary relationships.

8. The descriptions and differences between the workings of BLAST and PSI-BLAST are as follows :

a. BLAST

i. It searches one or more nucleic acid or protein databases for sequences similar to one or more query sequences of any type. We see that it uses similar instead of identical pairs.

ii. It uses a scoring matrix to score aligned pairs. Only those pairs that score above a threshold are considered for extension (which , in itself, is without gaps).

iii. Uncapped extension of HSPs with scores greater than the Threshold identifies maximal segment pairs

iv. The extension continues until the score drops below a threshold drop - off from the maximum score encountered. Thereafter, the highest scoring segment pair, the MSP is identified.

v. BLAST can produce gapped alignments for the matches it finds. This, it does using the same strategy as FASTA of joining segments on different diagonals.

b. PSI-BLAST
    i. It iteratively searches one or more protein databases for sequences similar to one or more protein query sequences. It is designed to find remote homologues with 15% - 25% identity levels.
    ii. It constructs scoring matrices by multiple alignment of hits obtained.
    iii. It searches the database with the new scoring matrix for every iteration. This iteration continues till convergence is reached.
    iv. The idea behind constructing a scoring matrix from the hits is that the new scoring matrix is tailor-made to find sequences similar to the query.

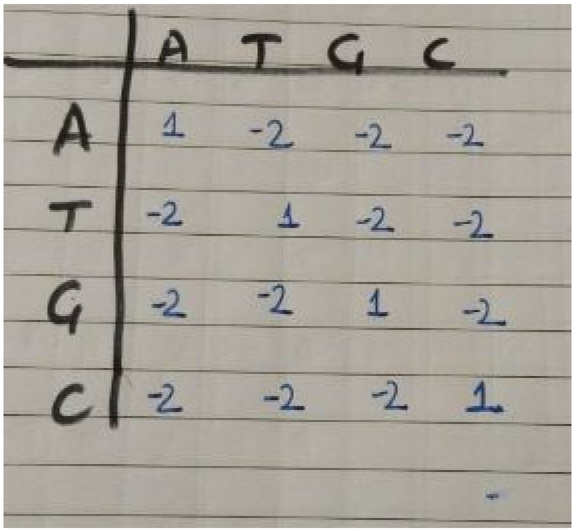
c. Now, although PSI-BLAST (also sometimes known as specialised BLAST) is almost similar to BLAST, the basic point of difference is that PSI-BLAST , unlike BLAST, uses position-specific scoring matrices derived during the search itself.

9. The solution for both the parts is as follows :

a. Scoring Matrices for match (M)/mismatch (N) ratio : Relative magnitudes of M & N determines the No. of nucleic acid PAMs (point accepted mutations per 100 residues) for which they are most sensitive at finding homologs. Therefore, the (absolute) reward/penalty ratio should be increased as one looks at more divergent sequences, and hence the match/mismatch ratio for comparing nucleotide sequences is chosen to be large for highly conserved sequences, while it is small for divergent sequences.

b. A match/mismatch ratio of 0.5(1 / -2) is best for sequences that are 95% conserved.

The matrix for the same is as follows :



	A	T	G	C
A	1	-2	-2	-2
T	-2	1	-2	-2
G	-2	-2	1	-2
C	-2	-2	-2	1

10. The solution is as written :

a. Substitution matrices for amino acids are more complicated and implicitly take into account everything that might affect the frequency with which any amino acid is substituted for another. The objective is to provide a relatively heavy penalty for aligning two residues together if they have a low probability of being homologous (correctly aligned by evolutionary descent).

b. The solution is yet to be writtenTwo major forces drive the amino-acid substitution rates away from uniformity: substitutions occur with the different frequencies, and lessen functionally tolerated than others. Thus, substitutions are selected against

c. For example, tryptophan and cysteine are often found at key positions in proteins where they play a critical role and cannot be substituted easily. Therefore, two aligned tryptophans get a high score (11) whereas two aligned alanines get a much lower score (4), because alanine residues can often be substituted quite readily by other amino acids.

d. Tryptophan is often found at key positions in proteins where they play a critical role and hence, they cannot be substituted easily. Therefore, two aligned tryptophans get a high score(11) in the substitution matrix. Also, these amino acids have unique chemistries and often play important structural or catalytic roles in proteins.

e. However, two aligned Leucine gets a much lower score (4), because Leucine residues can often be substituted quite readily by other amino acids.

11. We have to:

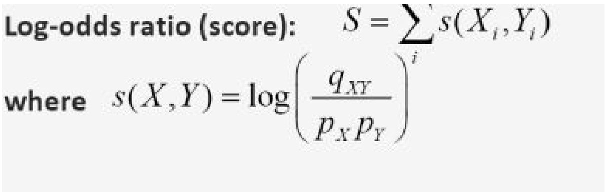
a. construct the scoring scheme for identifying DNA sequences that exhibit at least 65% identity.

b. Assume background frequency 0.25, for each of nucleotides and assume equiprobability for mismatch.

Equal probabilities for mismatch. Now from the above data, we can say that the self matches should occur with equal probabilities, that is 65% / 4 = 65/400 = 0.1625

And therefore the remaining pairs will be left with (0.25 - 0.1625)/3 = 0.03

score s(a, b) is given by :



Log-odds ratio (score): $S = \sum_i s(X_i, Y_i)$

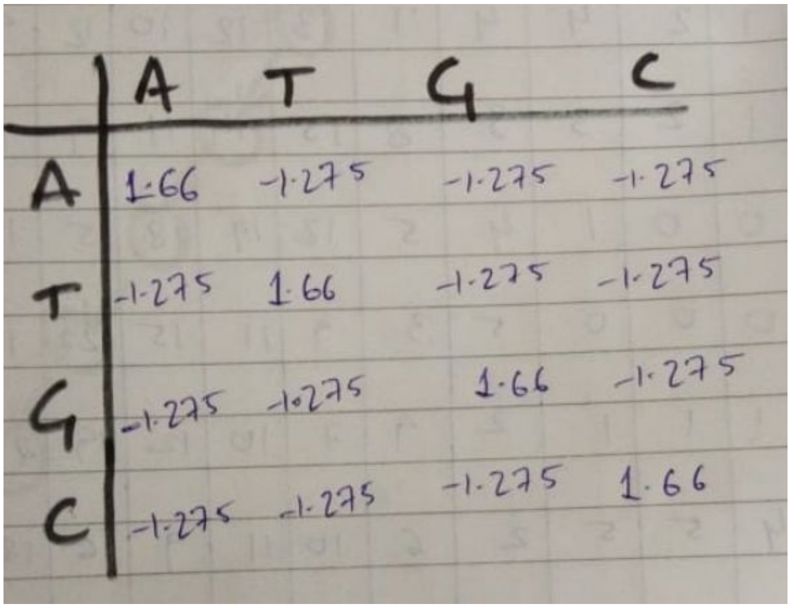
where $s(X,Y) = \log\left(\frac{q_{XY}}{p_X p_Y}\right)$

For s(a,a) : (1/0.25)* (log(0.1625/0.0625))

For s(a,b) : (1/0.25)*(log(0.03/0.0625))

Therefore scoring scheme is s(a,a)/s(a,b) = 1.66 /-1.275 = -1.305 = (+5)/(-4)

Therefore the matrix comes out to be : (note that the scores s(x,y) make the matrix entry for the position (X,Y)



	A	T	G	C
A	1.66	-1.275	-1.275	-1.275
T	-1.275	1.66	-1.275	-1.275
G	-1.275	-1.275	1.66	-1.275
C	-1.275	-1.275	-1.275	1.66