BMEG3105 Data analytics for personalized genomics and precision medicine
Student : WU Sio Fong
Sid:1155173201
Lecture 5: Assembly and mapping

Pre-course survey result:

Positive comment:	Criticism comment:	Lecture content question:
It was really taught well for this lecture. The algorithm of last pair was interesting to learn and using the table really helped a lot (was really good way to visualize how the algo would work in the computer) Clear ppt and speed Very interesting and interactive lecture illustrated by good example. I enjoy it	I think maybe: Cheapest flight problem was overexplained Could explain how to get the sub problems part more clearly Consider writing some steps on the how to get each alignment score in the matrix (mb the first two), show some steps for tracing back Would it be possible to show a snippet of a code to implement this kind of algorithm through the tutorial or just as supplementary material? It takes a long time before the lecture, started teaching the topic Question regarding the survey: Your understanding of the lecture content, I am not sure what you count as topic 1,2,3 ..	I want to know why we add extra gap in AGGC for example: in comparing The volume is a bit small, some words are unclear at the back Sequence Alignment with DP was explained confusingly, even though it is simple, and it seemed like many did not understand it

Dynamic programming (DP) [split problems into recursive sub-problem]
(building on the previous lesson)

- Purpose: identify sequence similarity between 2 sequences

- Analogy:
  Use fight finite destination as example→finite choice for base
  -Align to itself
  -Another base
  -Gap

- How we determine which choice is the best?
  → alignment score = $\sum score\ of\ each\ pair$

Scoring matrix:

	A	C	G	T
A	2	-7	-5	-7
C	-7	2	-7	-5
G	-5	-7	2	-7
T	-7	-5	-7	2

Score matrix is what we could define by our own, which will affect the result of the alignment
Depends on how we define the "similarity"!!!

- How we use DP to solve this problem based on alignment score matrix?
  Eg: F(ACCG,ACG) = Best[F(ACG,ACG) + F(G,__),
  F(ACCG,AC) + F(__,G),
  F(ACC,AC)+S(G,G)]
  and repeat doing this until we find the best solution

- Another way to show this process is through DP table (simplify the reduction preocess)
  Eg: To match ACCG & ACG

		A	C	C	G
	F(0,0)	F(1,0)	F(2,0)	F(3,0)	F(4,0)
A	F(0,1)	F(1,1)	F(2,1)	F(3,1)	F(4,1)
C	F(0,2)	F(1,2)	F(2,2)	F(3,2)	F(4,2)
G	F(0,3)	F(1,3)	F(2,3)	F(3,3)	F(4,3)

Figure 1. Position of each cell

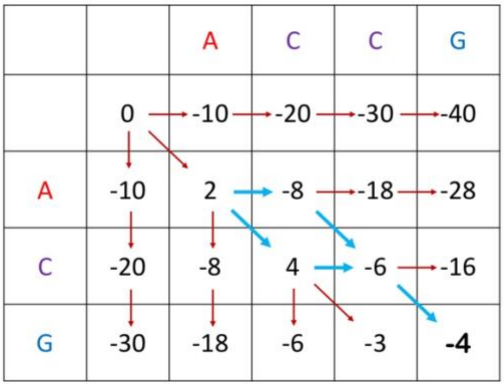


Figure2. Show best value of each combination

- Elements in the table:
  - Value: -Fill the table according to the matrix :go right/down/ diagonal right
    -Each cell keep max value from above calculation
    -If the value is small enough, we don't calculate anymore
    -We only focus on the last cell value (Global and local alignment *)

  - Arrow: Showing the path; 2 path →2 optimal alignments
    Trace back to find the alignment after filling
    The alignment for ACCG & ACG are: ACCG,A_CG; ACCG,AC_G

*Addition:
Global alignment→ to find similar components, motifs and domains and in dissimilar sequences
Local alignment→to compare 2 different sequence similarity

- Time complex of DP : $O(n^2)$ [which is better than enumeration result : $\frac{(2n)!}{(n!)(n!)}$]

- Corresponding real biology situation:

Score matrix	Reality meaning in biology
Mismatch	Mutations
Gap	Insertion/deletion; gene duplications

- Additional Source:
  Webserver for sequence alignment: https://www.ebi.ac.uk/Tools/psa/emboss_needle/
  Biopython: https://biopython.org

Sequence Data

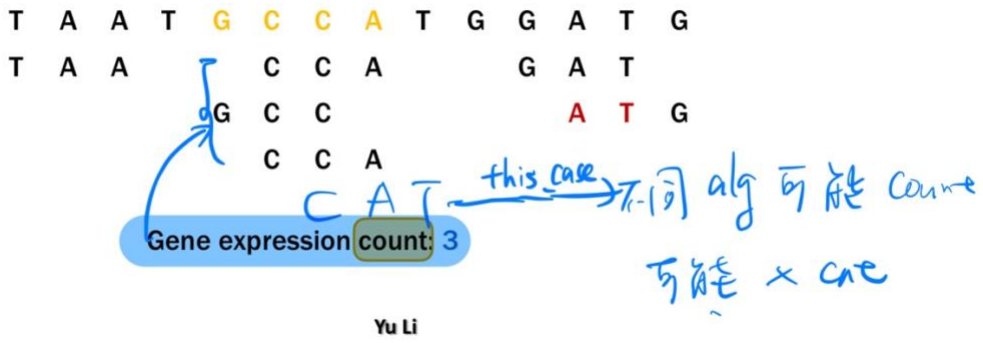
- Why sequence Data
  - Central dogma
  - Hidden genetic information
  - Phenotype = genotype (sequence) + environment
  - Human genome is mostly same (0.001% variation)
  - 1% of the genome control encoding protein

- How we get gene expression matrix from sequence?
  - mRNA isolation from sample
  - illumine sequencing in machine
  - align sequences against genome to construct the matrix
    -map the short read to the genome
    -count the number of read -- > gene expression matrix

- **Genome assembly**

  - Purpose: reconstruct the complete DNA sequence of an organism's genome
  - Possible problem:
    -mutation, conflict (due to noise/error)
    -repeated sequences; repeat genes; faster algorithm
    →Possible solution: do a longer sequence

  - Mapping example:
    slide the read along the genome & calculate the difference



- Resource and Uncover Part
  - Bioinformatics: Sequence and Genome Analysis---Chapter 2&3
  - Time complexity and space complexity analysis
  - Local alignment
  - Multiple sequence alignment

- Affine gap penalty
- Sequence database search: BLAST