

22.08 Algoritmos y Estructuras de Datos

Level 5: DNATool

Introducción

En esta práctica vamos a alinear secuencias genéticas de coronavirus (SARS-CoV-2).

Genetics 101

El *ácido desoxirribonucleico*, conocido también por las siglas ADN, contiene las instrucciones genéticas usadas en el desarrollo y funcionamiento de todos los organismos vivos.

Es un polímero de nucleótidos, en el que cada nucleótido está formado por una base nitrogenada (*adenina*, *citosina*, *guanina* o *timina*), un glúcido y un grupo fosfato.

Desde el punto de vista bioinformático, el ADN es una secuencia de símbolos “A”, “C”, “G” y “T”.

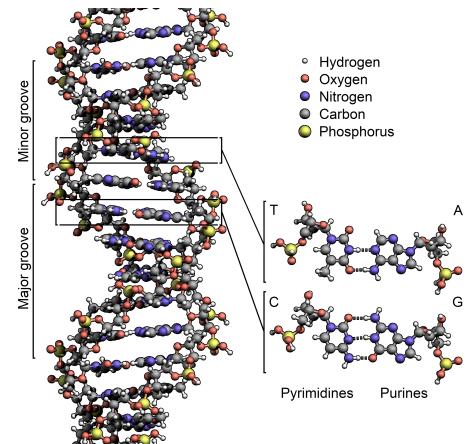


Fig. 1 - Modelo de ADN

El formato GenBank

En esta práctica vamos a trabajar con secuencias genéticas en formato *GenBank*, un formato basado en texto con una estructura muy simple.

Veamos un ejemplo de un archivo GenBank:

```
LOCUS      NM_000518          626 bp   mRNA    linear    PRI 01-MAY-2018
DEFINITION Homo sapiens hemoglobin subunit beta (HBB), mRNA.
ACCESSION NM_000518
VERSION   NM_000518.4
KEYWORDS  RefSeq.
SOURCE    Homo sapiens (human)
ORGANISM  Homo sapiens
                      Eukaryota; Metazoa; Chordata; Craniata; Vertebrata; Euteleostomi;
                      Mammalia; Eutheria; Euarchontoglires; Primates; Haplorrhini;
                      Catarrhini; Hominidae; Homo.
ORIGIN
       1 acatggctt ctgacacaac tgtgttcaact agcaacacct aacagacacc atgggtcatc
       61 tgactcctga ggagaagtct gccgttactg ccctgtgggg caaggtaaac gtggatgaag
```

```
121 ttggtgtga ggcctggc aggctgctgg tggctaccc ttggacccag agttcttg  
181 agtccttgg gatatctgtcc actcctgatg ctgttatggg caaccctaag gtgaaggctc  
241 atggcaagaa agtgctcggt gccttagtg atggcctggc tcacctggac aacctcaagg  
301 gcaccttgc cacactgagt gagctgact gtgacaagct gcacgtggat cctgagaact  
361 tcaggctcct gggcaacgtg ctggctgtg tgctggccca tcacttggc aaagaattca  
421 ccccaccagt gcaggctgcc tatcagaaag tggtggtgg tggctaat gccctggccc  
481 acaagtatca ctaagctcgc tttcttgctg tccaatttct attaaagggtt cttttgttcc  
541 ctaagtccaa ctactaaact ggggatatt atgaagggcc ttgagcatct ggattctgcc  
601 taataaaaaa catttatttt cattgc  
//
```

La parte del archivo que vamos a necesitar es la secuencia genética cruda, que se encuentra entre la línea **ORIGIN** y la línea **//** (doble barra). La secuencia genética está representada por los caracteres: “a” para adenina, “c” para citosina, “g” para guanina y “t” para timina. Todos los demás caracteres se descartan.

Alineamiento de secuencias

Un *alineamiento de secuencias* es una representación de dos o más secuencias genéticas que resalta sus zonas de similitud, lo cual podría indicar relaciones funcionales o evolutivas.

Las secuencias alineadas se representan típicamente en filas de una matriz, en las que, si es necesario, se insertan espacios para que las zonas con idéntica o similar estructura se alineen.

A modo de ejemplo, intentemos alinear estas dos secuencias de ADN:

GATTACA¹

CGATACG

Un posible alineamiento es:

-GATTACA
 ||| ||*
CGAT-ACG

En este alineamiento hay una *inserción* al principio, una *deleción* en el cuarto carácter, y una *sustitución* en el último carácter.

Solemos indicar las *coincidencias* con ‘|’, los *indel* (inserciones y delecciones) con ‘ ’ (espacio), y las sustituciones con ‘*’.

¹ Es también el nombre de una famosa película.

El algoritmo de Needleman-Wunsch

Para alinear automáticamente dos secuencias A y B vamos a usar el *algoritmo de Needleman-Wunsch*, que fue desarrollado por Saul B. Needleman y Christian D. Wunsch en 1970.

El algoritmo asigna un puntaje a cada posible alineamiento, y busca aquellos alineamientos que maximizan el puntaje. En esta práctica vamos a calcular los puntajes con el siguiente *sistema de puntaje*:

- `matchScore` (coincidencia): +1
- `substScore` (sustitución): -1
- `indelScore` (inserción o delección): -1

Por ejemplo, para el alineamiento de antes:

```
-GATTACA
   ||| ||*
CGAT-ACG
```

El puntaje es:

$$5 \text{ (coincidencias)} - 1 \text{ (sustituciones)} - 2 \text{ (indel)} = 2$$

Para encontrar el alineamiento óptimo definimos una matriz \mathbf{F} de $m + 1$ por $n + 1$ elementos, en donde m y n son el tamaño de las secuencias A y B , respectivamente.

Cada celda $\mathbf{F}_{i,j}$ de la matriz representa el puntaje del alineamiento óptimo entre los primeros i elementos de A y los primeros j elementos de B .

Con el ejemplo de antes:

	C	G	A	T	A	C	G
G							
A							
T							
A							
C							
A							

Para determinar el puntaje de cada celda $\mathbf{F}_{i,j}$ calculamos puntajes desde la celda vecina superior $\mathbf{F}_{i-1,j}$, desde la superior-izquierda $\mathbf{F}_{i-1,j-1}$ y desde la izquierda $\mathbf{F}_{i,j-1}$:



- El camino desde arriba representa una delección. El *puntaje de delección* es el puntaje de la celda superior $\mathbf{F}_{i-1,j}$ más `indelScore`.
- El camino diagonal representa una coincidencia o sustitución. El *puntaje de coincidencia/sustitución* es el puntaje de la celda superior-izquierda $\mathbf{F}_{i-1,j-1}$, más, en caso de coincidencia, `matchScore`, o, en caso de sustitución, `substScore`.
- El camino desde la izquierda representa una inserción. El *puntaje de inserción* es el puntaje de la celda izquierda $\mathbf{F}_{i,j-1}$ más `indelScore`.

En la celda $\mathbf{F}_{i,j}$ guardamos el máximo de estos tres puntajes. También almacenamos cuál(es) direcciones maximizan el puntaje.

Inicializamos el algoritmo con $\mathbf{F}_{0,0} = 0$. Para la primera fila de la matriz sólo hay caminos desde la izquierda, por tanto: $\mathbf{F}_{0,j} = \mathbf{F}_{0,j-1} + \text{indelScore}$. En forma análoga, para la primera columna: $\mathbf{F}_{i,0} = \mathbf{F}_{i-1,0} + \text{indelScore}$.

En nuestro ejemplo:

	C	G	A	T	A	C	G	
0	0	-1	-2	-3	-4	-5	-6	-7
G	-1							
A	-2							
T	-3							
T	-4							
A	-5							
C	-6							
A	-7							

Ahora completamos las celdas, fila a fila, de arriba a abajo. Resultará:

	C	G	A	T	A	C	G	
0	0	-1	-2	-3	-4	-5	-6	-7
G	-1	-1	0	-1	-2	-3	-4	-5
A	-2	-2	-1	1	0	-1	-2	-3
T	-3	-3	-2	0	2	1	0	-1
T	-4	-4	-3	-1	1	1	0	-1
A	-5	-5	-4	-2	0	2	1	0
C	-6	-4	-5	-3	-1	1	3	2
A	-7	-5	-5	-4	-2	0	2	2

Podemos encontrar el puntaje de la alineación óptima entre la secuencia A y B en la celda $\mathbf{F}_{m,n}$. Para este ejemplo, este puntaje es 2.

Para encontrar un camino óptimo, sólo basta invertir las flechas y rastrear un camino desde $\mathbf{F}_{m,n}$ a $\mathbf{F}_{0,0}$:

	C	G	A	T	A	C	G	
	0	-1	-2	-3	-4	-5	-6	-7
G	-1	-1	0	-1	-2	-3	-4	-5
A	-2	-2	-1	1	0	-1	-2	-3
T	-3	-3	-2	0	2	1	0	-1
T	-4	-4	-3	-1	1	1	0	-1
A	-5	-5	-4	-2	0	2	1	0
C	-6	-4	-5	-3	-1	1	3	2
A	-7	-5	-5	-4	-2	0	2	2

A partir de este camino podemos determinar el alineamiento, insertando sendos espacios en donde el recorrido sigue un camino horizontal o vertical:

```
-GATTACA
 ||| ||*
 CGAT-ACG
```

Pueden haber múltiples caminos que produzcan el mismo puntaje óptimo. De hecho, el ejemplo anterior tiene dos caminos óptimos. ¿Puedes encontrar el otro?

Más sobre el algoritmo de Needleman-Wunsch

El algoritmo de Needleman-Wunsch es muy usado en análisis genético, en análisis financiero, en procesamiento de lenguaje natural, en procesamiento de voz, en identificación de música y en visión artificial.

Es uno de los primeros en utilizar el *método de la programación dinámica*, que resuelve un problema complejo separándolo en sub-problemas más simples.

Esta idea muy poderosa permite resolver muchos problemas algorítmicos en forma eficiente.

A trabajar

Deberás escribir un programa que recibe por línea de comando los nombres de dos archivos GenBank a ser alineados. El programa deberá imprimir en consola el puntaje óptimo y un alineamiento óptimo para ambas secuencias, tal como te mostramos en la sección *Alineamiento de secuencias*. Si hay múltiples alineamientos óptimos, elige uno cualquiera. Divide el alineamiento en líneas de 60 caracteres.

Para guardar tus resultados a disco, aprovecha la redirección de la línea de comando:

```
[tu archivo ejecutable] [archivo1] [archivo2] > [archivo de salida]
```

Recomendaciones:

- Escribe una función para leer archivos GenBank, y otra para determinar el alineamiento; en este problema no tiene sentido usar clases.
- Haz programación orientada a pruebas.
- Aprovecha la función `reverse` de STL.
- Para saber si estás haciendo las cosas bien, aprovecha [esta herramienta de la universidad de Freiburg](#). Cuidado: debes cambiar el valor “gap” a -1 para que coincida con nuestro sistema de puntaje.

Tareas

Junto a este enunciado encontrarás un *dataset de referencia* que contiene la secuencia genética del SARS-CoV-2 original ([Wuhan reference genome – NC_045512.2.txt](#)) y la de seis variantes (alpha, beta, gamma, delta, omicron BA.1 y omicron BA.2).

También encontrarás un *dataset de test*, que contiene tres “mystery genomes”. Usando tu programa, deberás determinar, para cada uno de ellos, a qué genoma de referencia se parece más, y qué diferencias tiene respecto de éste.

Entrega tus respuestas en sendos archivos `.txt`.

Bonus points

- ¿Cuál es la complejidad computacional del algoritmo de Needleman-Wunsch?
- Prueba otros valores para el sistema de puntaje. ¿Qué observas?
- Compara otras secuencias genéticas. Puedes encontrar más secuencias en la [National Library of Medicine](#).