**Universidad de Buenos Aires (FIUBA)**

Maestría en Inteligencia Artificial

# Trabajo Práctico 3

## Needleman–Wunsch (Alineación Global)

**Materia:** Computación, Algoritmos y Estructuras de Datos
**Docente:** Dr. Lic. Camilo Argoty
**Alumna:** Esp. Lic. Noelia Qualindi
**SIU:** a1411

Repositorio del trabajo:

`https://github.com/noequalindi/computing_algorithms/tree/main/tp3`

# 1. Parte 1: Conceptos teóricos

## 1.1. Secuencias de nucleótidos

Una secuencia de nucleótidos es una cadena sobre un alfabeto finito, típicamente $\{A, C, G, T\}$ para ADN (o $\{A, C, G, U\}$ para ARN). Compararlas permite identificar similitudes, mutaciones (sustituciones, inserciones, deleciones) y relaciones evolutivas.

## 1.2. Alineación de secuencias

Un alineamiento inserta huecos (gaps) "-" para poner en correspondencia posiciones entre dos secuencias.

- **Alineación global**: alinea de extremo a extremo (se fuerza cubrir toda la longitud).
- **Alineación local**: busca el segmento más similar (no necesariamente usa toda la secuencia).

En un alineamiento aparecen:

- *matches*: misma letra,
- *mismatches*: letras distintas,
- *gaps*: inserciones/deleciones modeladas con "-".

## 1.3. Modelo de puntuación

Se usa un esquema simple:

- Match: $+1$
- Mismatch: $-1$
- Gap: $-2$

(En modelos más realistas, se distinguen penalizaciones de apertura/extensión de gap; aquí se usa penalización constante por gap.)

## 1.4. Algoritmo de Needleman–Wunsch

Needleman–Wunsch resuelve alineación global mediante programación dinámica:

- Construye una matriz $F$ de tamaño $(n+1) \times (m+1)$.
- Inicializa primera fila/columna acumulando gaps.
- Usa la recurrencia:

$$F[i,j] = \text{máx} \begin{cases} F[i-1, j-1] + s(x_i, y_j) \\ F[i-1, j] + \text{gap} \\ F[i, j-1] + \text{gap} \end{cases}$$

- Luego aplica *traceback* desde $F[n, m]$ para recuperar un alineamiento óptimo.

El puntaje óptimo de alineación global queda en la celda inferior derecha $F[n, m]$.

### 1.5. Referencia

Needleman, S. B., & Wunsch, C. D. (1970). *A general method applicable to the search for similarities in the amino acid sequence of two proteins.* Journal of Molecular Biology, 48(3), 443–453.

## 2. Parte 2: Implementación

Se implementó desde cero Needleman–Wunsch en Python con el esquema de puntuación indicado. Para cada par de secuencias el programa imprime:

- matriz completa de puntuación,
- un alineamiento global óptimo,
- puntaje final.

### 2.1. Cómo correr

```
cd tp3
python tp3_needleman_wunsch.py --examples
```

## 3. Evidencia de ejecución (salidas del programa)

### 3.1. Ejecución con tres parejas

A continuación se muestra la salida obtenida al ejecutar el comando `-examples`. Se incluyen las matrices completas, los alineamientos óptimos y el puntaje final de cada caso.

```
python tp3_needleman_wunsch.py --examples

============================================================
Sequence 1: GATTACA
Sequence 2: GCATGCU
Score matrix:
      _   G   C   A   T   G   C   U
  _   0  -2  -4  -6  -8 -10 -12 -14
  G  -2   1  -1  -3  -5  -7  -9 -11
  A  -4  -1   0   0  -2  -4  -6  -8
  T  -6  -3  -2  -1   1  -1  -3  -5
  T  -8  -5  -4  -3   0   0  -2  -4
  A -10  -7  -6  -3  -2  -1  -1  -3
  C -12  -9  -6  -5  -4  -3   0  -2
  A -14 -11  -8  -5  -6  -5  -2  -1

Optimal alignment (global):
GATTACA
GCATGCU
Total score: -1
============================================================

============================================================
Sequence 1: ACGT
Sequence 2: ACCT
Score matrix:
    _  A  C  C  T
 _  0 -2 -4 -6 -8
 A -2  1 -1 -3 -5
 C -4 -1  2  0 -2
 G -6 -3  0  1 -1
 T -8 -5 -2 -1  2

Optimal alignment (global):
ACGT
ACCT
Total score: 2
============================================================

============================================================
Sequence 1: ATGCT
Sequence 2: AGCT
Score matrix:
      _   A   G   C   T
  _   0  -2  -4  -6  -8
  A  -2   1  -1  -3  -5
  T  -4  -1   0  -2  -2
  G  -6  -3   0  -1  -3
  C  -8  -5  -2   1  -1
  T -10  -7  -4  -1   2
```

```
Optimal alignment (global):
ATGCT
A-GCT
Total score: 2
============================================================
```

### 3.2. Verificación del puntaje del Caso 1

En el Caso 1 se obtuvo un alineamiento sin gaps:

`GATTACA` vs `GCATGCU`.

Con match $= +1$ y mismatch $= -1$, el puntaje total es:

$$(+1) + (-1) + (-1) + (+1) + (-1) + (+1) + (-1) = -1,$$

lo cual coincide con el valor reportado por el algoritmo y con la celda $F[n, m]$ de la matriz de puntuación.

### 3.3. Repositorio

El código fuente y salidas reproducibles se entregan vía repositorio:

`https://github.com/noequalindi/computing_algorithms/tree/main/tp3`

## 4. Bibliografía

## Referencias

[1] S. B. Needleman and C. D. Wunsch. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology*, 48(3):443–453, 1970.

[2] Richard Durbin, Sean Eddy, Anders Krogh, Graeme Mitchison. *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*. Cambridge University Press, 1998. (Algoritmos de alineación y programación dinámica).