

Introducción a la Bioinformática

Alineamiento de secuencias

Búsqueda de secuencias en bases de datos

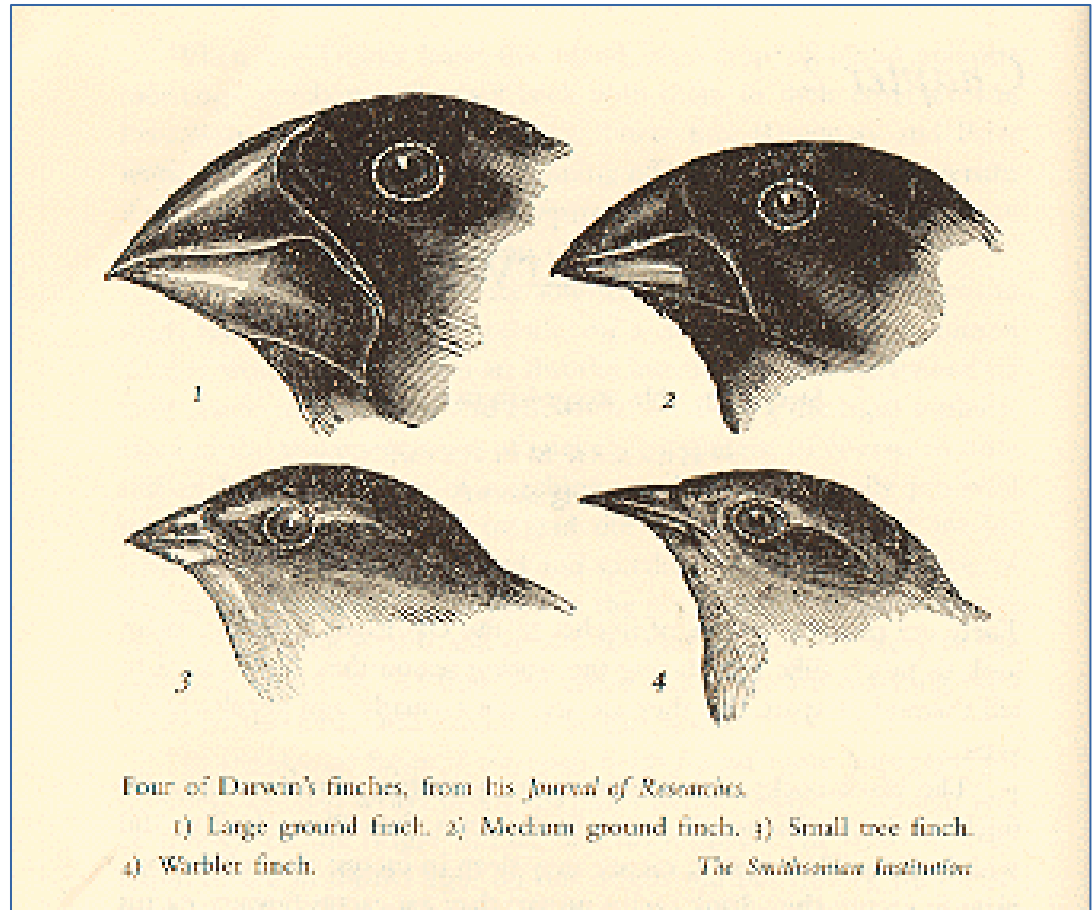
Fernán Agüero

**Instituto de Investigaciones Biotecnológicas
Universidad Nacional de San Martín**

Análisis comparativo

El alineamiento de secuencias es similar a otros tipos de análisis comparativo.

En ambos es necesario cuantificar las similitudes y diferencias (scoring) entre un grupo relacionado de entidades.



Finches of the Galápagos Islands observed by Charles Darwin on the voyage of HMS *Beagle*

Alinear secuencias

Para poder comparar
secuencias, tenemos que
sistematizar la manera en que
lo hacemos

Por donde empezamos?

Comparamos las dos
secuencias letra a letra,
empezando por la primera?
Tiene sentido?

GCTACTAGTTCGCTTAGC

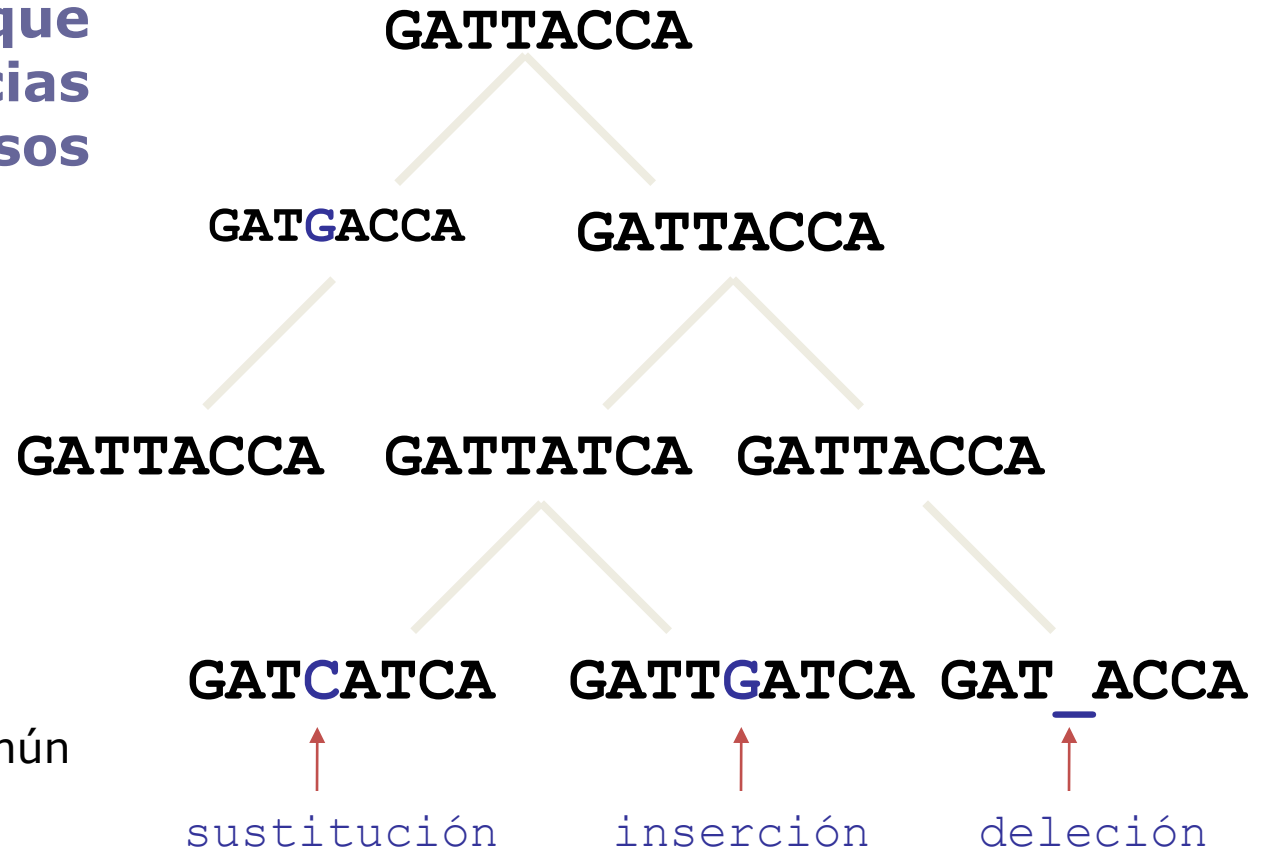
GCTACTAGCTCTAGCGCGTATAGC

Homología vs similitud

- Homología entre dos entes biológicos implica una herencia compartida
- Homología es un término cualitativo
- Se es homólogo o no se es
- Similitud implica una apreciación cuantitativa o una cuantificación directa de algún carácter
- Podemos usar una medida de similitud para **inferir** homología

Análisis comparativo

Los algoritmos que
alinean secuencias
modelan procesos
evolutivos

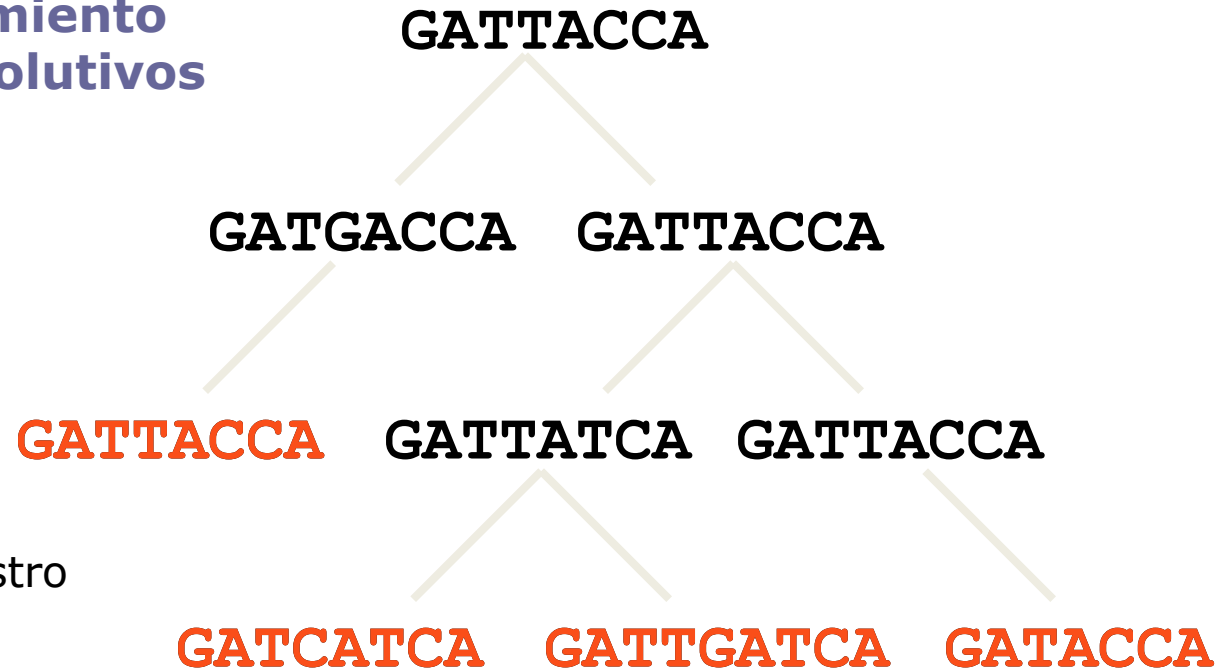


Deriva de un ancestro común a través de cambios incrementales debido a errores en la replicación del DNA, mutaciones, daño o crossing-over desigual.

Análisis comparativo

**Algoritmos de alineamiento
modelan procesos evolutivos**

Deriva a partir de un ancestro
común a través de cambio
incremental.

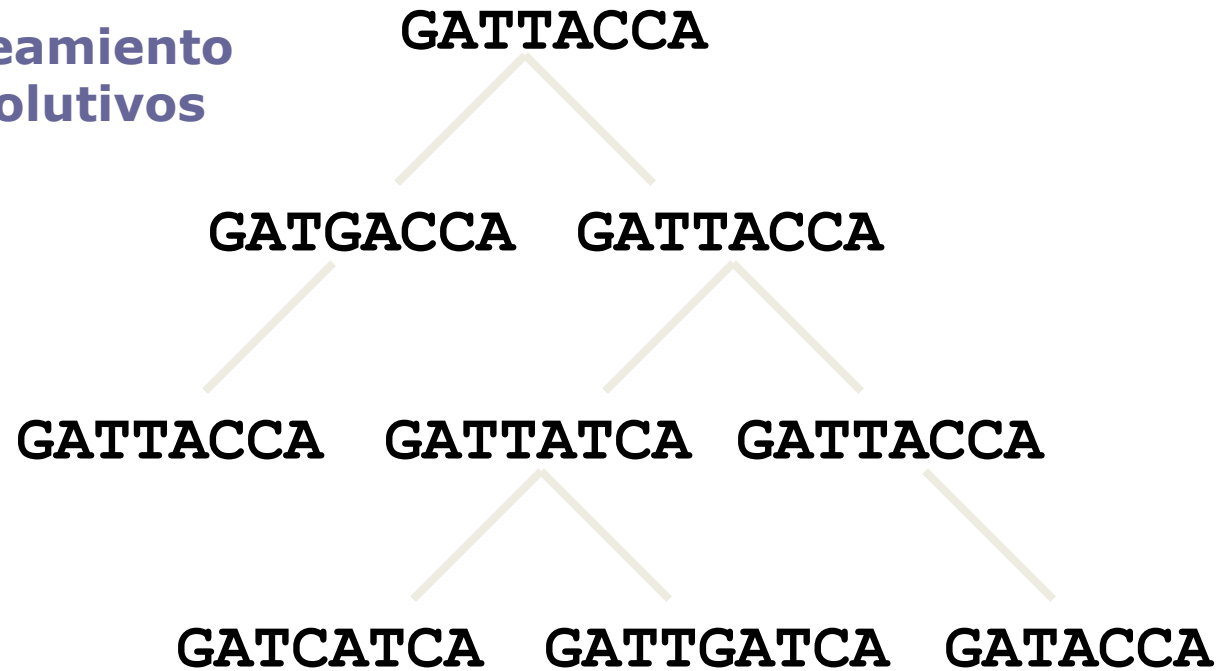


**Sólo las secuencias actuales son conocidas, las
secuencias ancestrales se postulan.**

Análisis comparativo

Algoritmos de alineamiento modelan procesos evolutivos

Deriva a partir de un ancestro común a través de cambio incremental.
Mutaciones que no matan al individuo pueden pasar a la población.



La palabra **homología** implica una herencia común (un ancestro común), el cual puede ser inferido a partir de observaciones de **similitud** de secuencia.

- **Qué es un alineamiento?**
 - **El procedimiento de comparación de dos (o más) secuencias que busca una serie de caracteres individuales o patrones de caracteres que se encuentren en el mismo orden en ambas secuencias**
- **Cómo alineamos dos secuencias?**
 - **Usando un método (algoritmo)**
 - a mano (como en los viejos tiempos)
 - usando una computadora

Definición de alineamiento: tipos

- Alineamiento:** Cada base se usa a lo sumo una vez
- Alineamiento global:** Todas las bases se alinean con otra base o con un gap ("-")
- Alineamientos locales:** No hay necesidad de alinear todas las bases

Align **GATESLIKESCHEESE** and **GRATEDCHEESE**

G-ATESLIKESCHEESE	or	G-ATES	& CHEESE
GRATED-----CHEESE		GRATED	& CHEESE

Alineamientos buenos y malos?

Cuál es el 'mejor' alineamiento?

GCTACTAG-T-T--CGC-T-TAGC
GCTACTAGCTCTAGCGCGTATAGC

0 mismatches, 5 gaps

GCTACTAGTT-----CGCTTAGC
GCTACTAGCTCTAGCGCGTATAGC

3 mismatches, 1 gap

Cómo decidir cuál es el mejor?

- Respuesta: el más significativo desde el punto de vista biológico
- Pero: necesitamos una medida **objetiva**
- **sistemas de puntaje (scoring)**
 - reglas para asignar puntos
 - el más simple: match, mismatch, gap

Un primer sistema de puntajes

Ejemplo de sistema de score

match = +1

mismatch = 0

gap = -1

G-ATESLIKESCHEESE
GRATED-----CHEESE

Score

$$(10 * 1) + (1 * 0) + (6 * (-1)) = +4$$

Cambiamos nuestro sistema de puntajes

Usando otro de sistema de puntajes?

match = +2

mismatch = 0

gap = -1

G-ATESLIKESCHEESE
GRATED-----CHEESE

Usando otro sistema de score

Score

$$(10 * 2) + (1 * 0) + (5 * (-1)) = +14$$

No se pueden comparar scores

- **Primera conclusión importante:**
 - **no tiene sentido comparar scores de distintos alineamientos**
 - **a menos que se especifique el sistema de scoring utilizado**

Gap penalties

gap opening penalty = -5

gap extension penalty = -1

1- Abrir un gap es costoso

GCTACTAG-T-T--CGC-T-TAGC
GCTACTAGCTCTAGCGCGTATAGC

$$\text{Penalty} = 5 * (-5) + 6 * (-1) = -31$$

2 - Extender un gap es menos costoso

GCTACTAGTT-----CGCTTAGC
GCTACTAGCTCTAGCGCGTATAGC

$$\text{Penalty} = 1 * (-5) + 6 * (-1) = -11$$

Dot plots: introducción

Dot-plot: Fitch, Biochem. Genet. (1969) 3, 99-108.

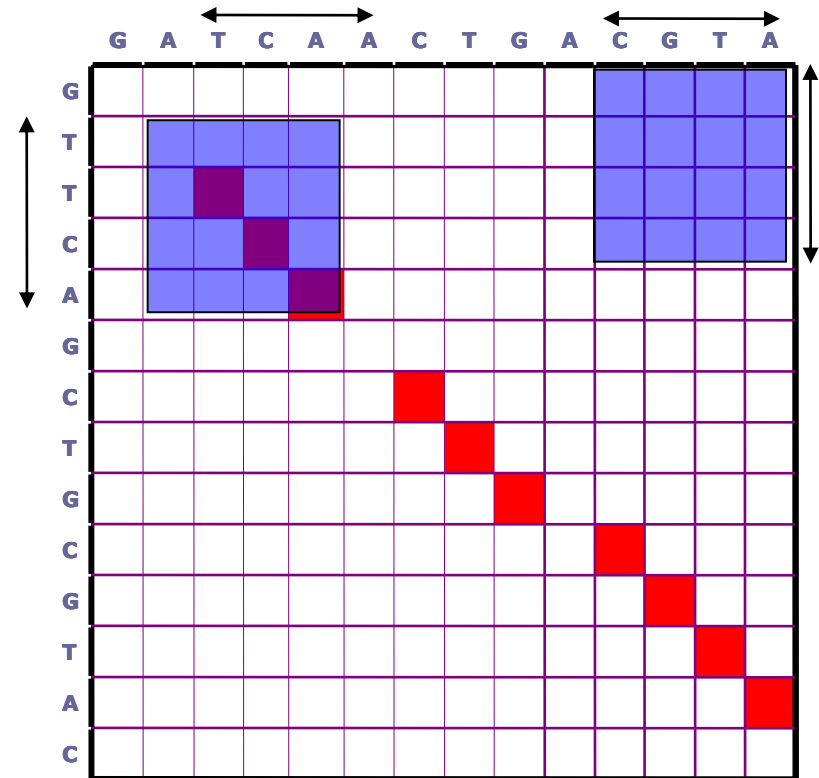
Eje horizontal: secuencia 1

Eje vertical: secuencia 2

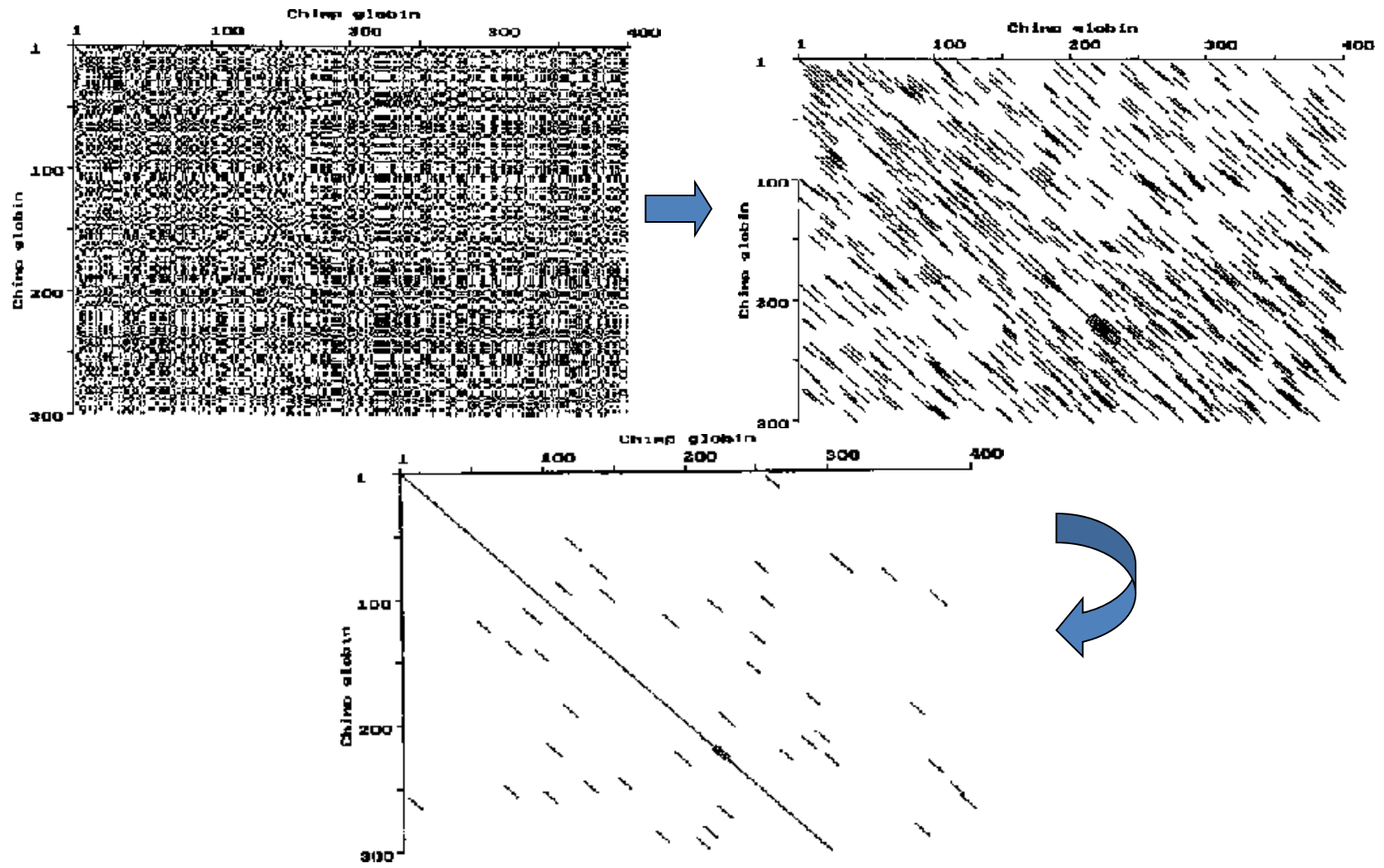
	C	G	T	A	C	C	G	T
A	0	0	0	1	0	0	0	0
C	1	0	0	0	1	1	0	0
G	0	1	0	0	0	0	1	0
T	0	0	1	0	0	0	0	1

Dot Matrix Plot

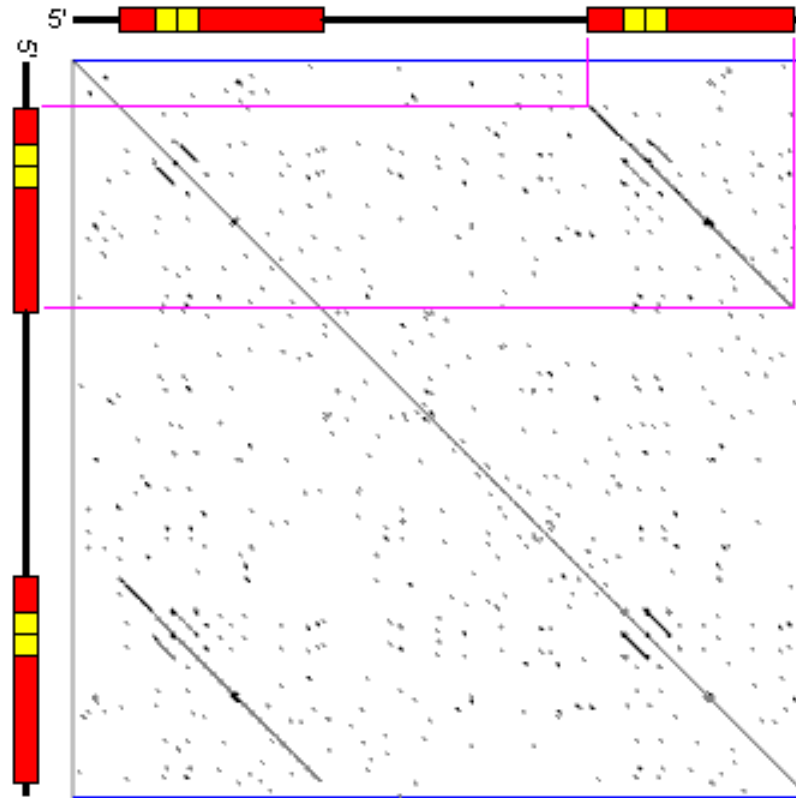
- Dos secuencias, una vertical y otra horizontal a los ejes del gráfico.
- Se colocan “puntos” en donde hay un match.
- Las líneas diagonales son regiones de identidad.
- Se aplican filtros para mejorar la comprensión del gráfico.



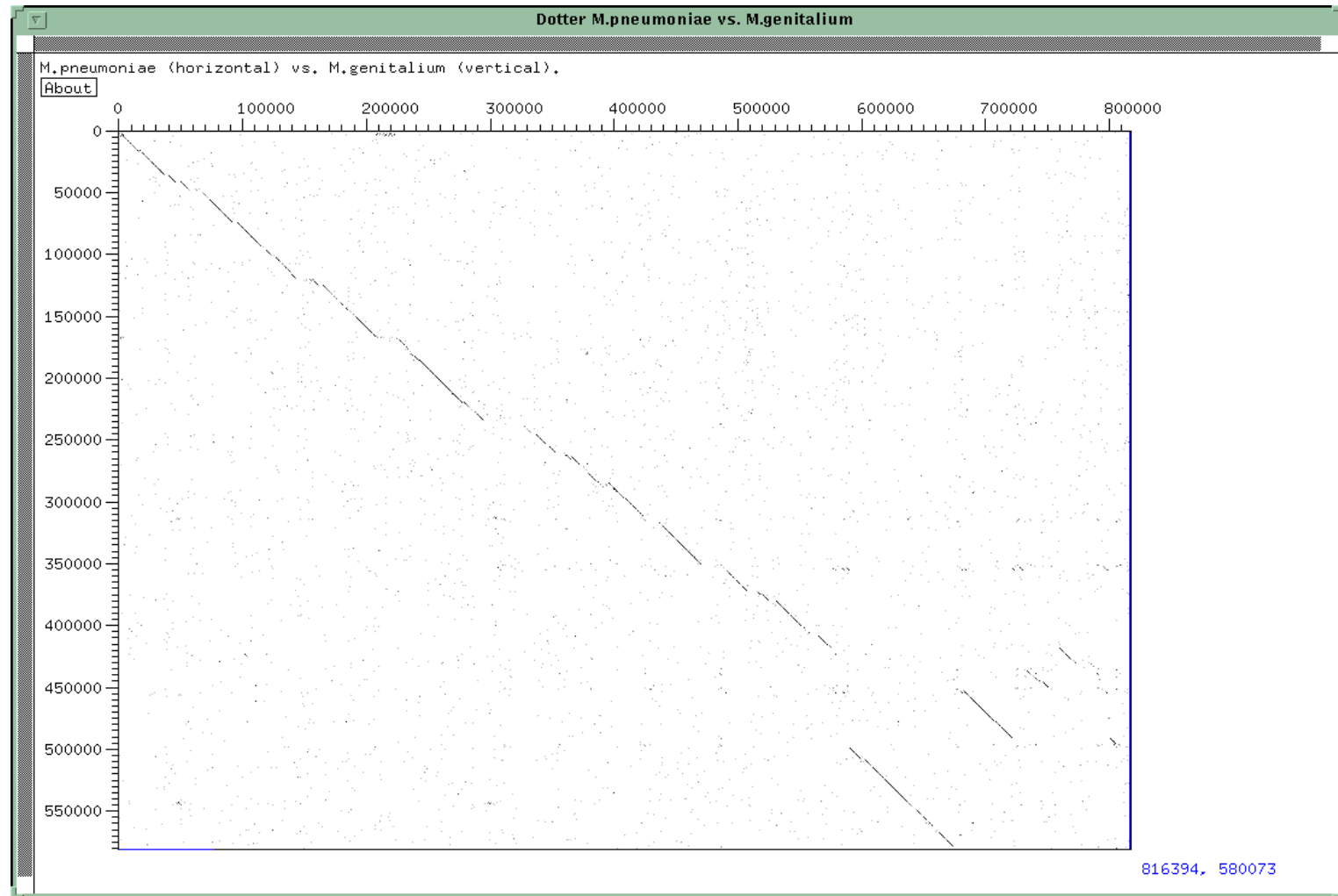
Dot Matrix Plot



Dot Matrix Plot



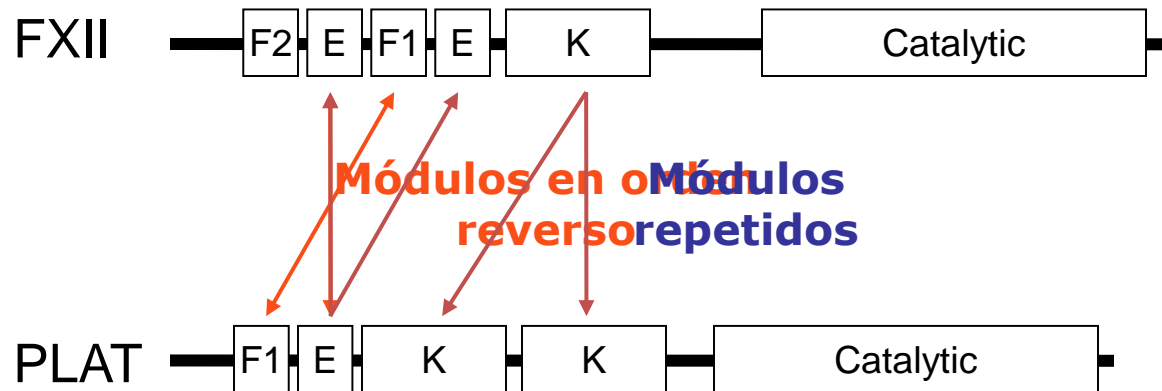
Dot Matrix Plot



Similitud local

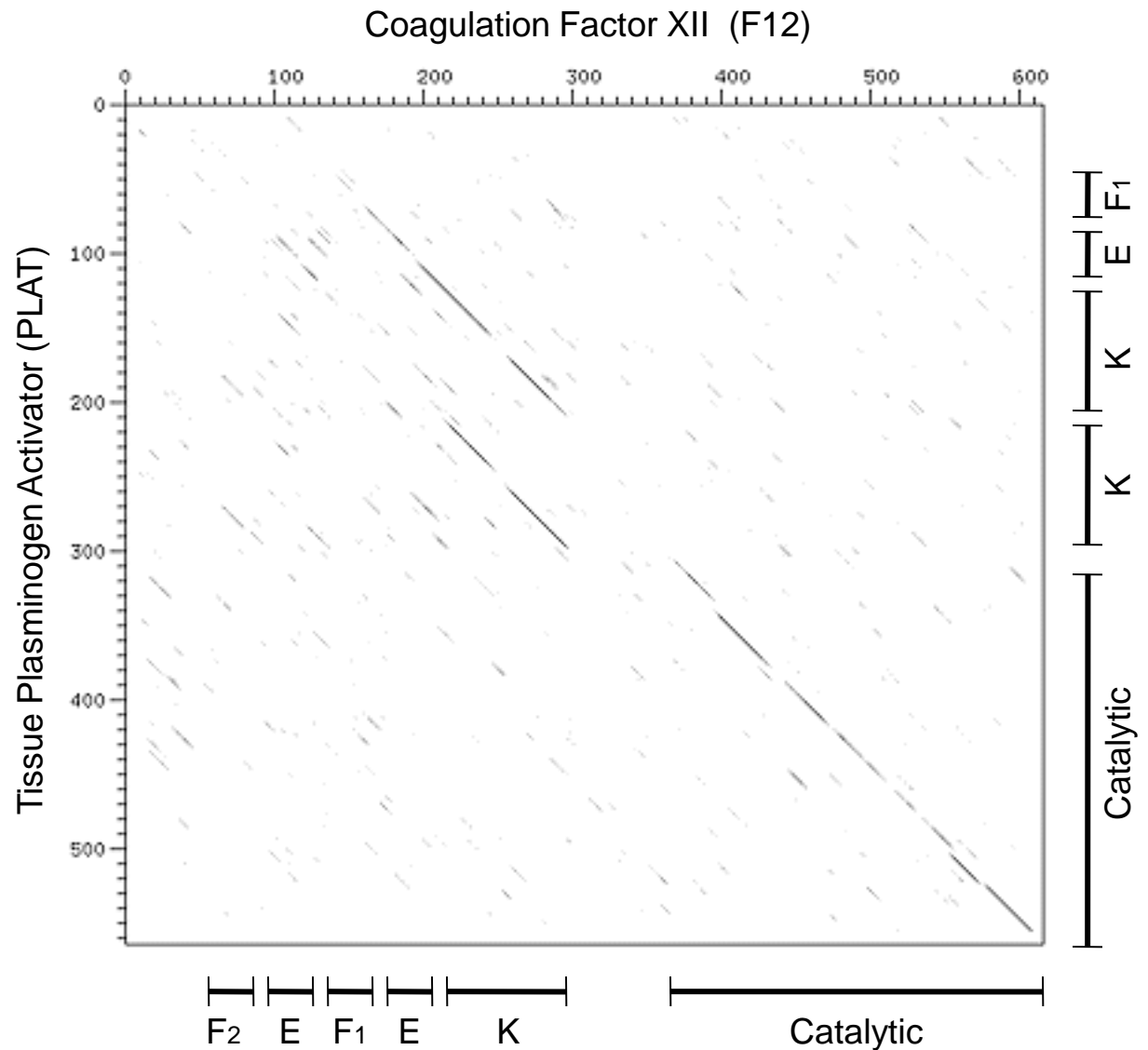
Dominios mezclados confunden a los algoritmos de alineamiento.

Módulos en el factor XII de coagulación y en el activador de plasminógenos – tissue plasminogen activator (PLAT)



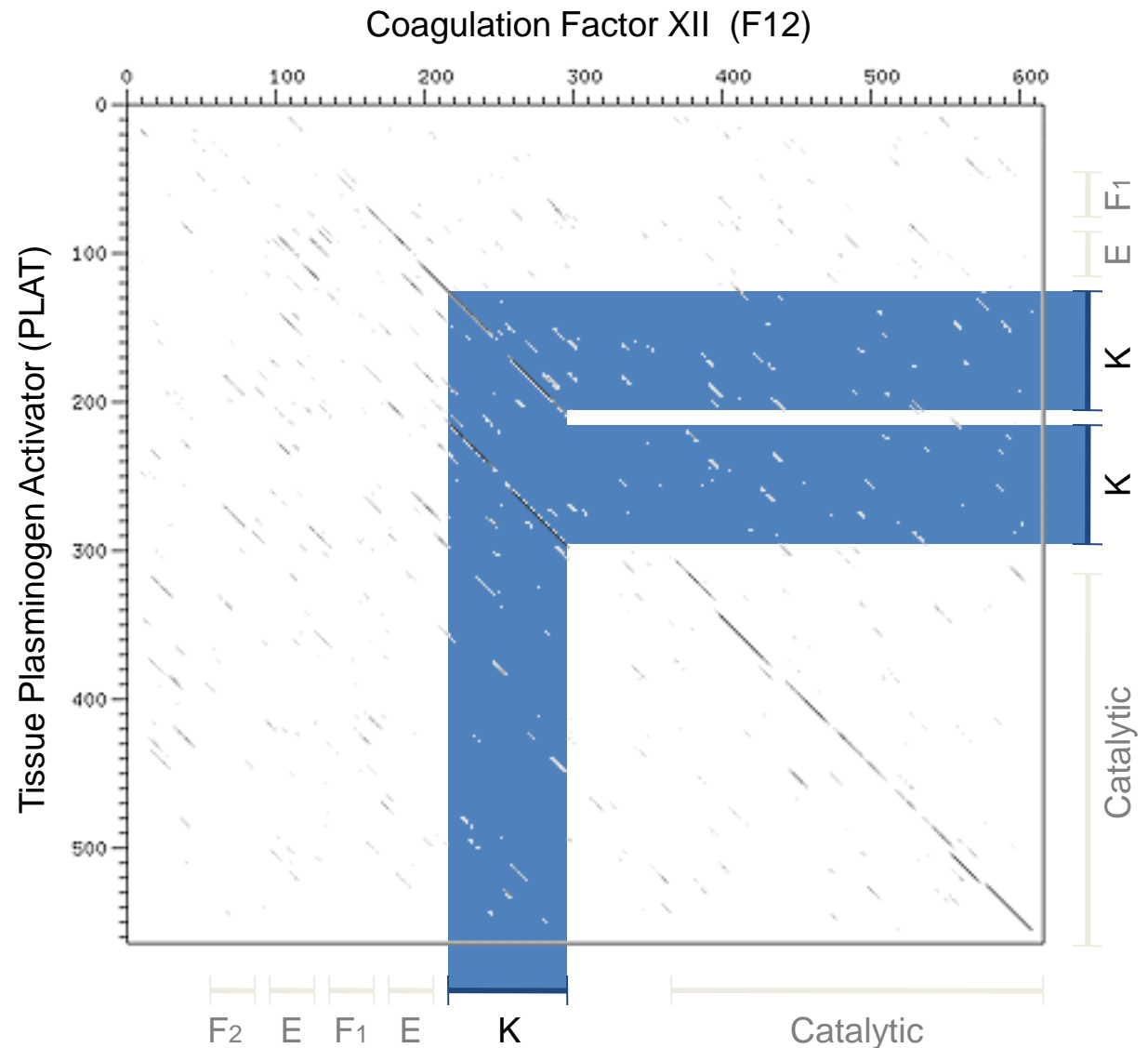
F1, F2	Fibronectin repeats
E	EGF similarity domain
K	Kringle domain
Catalytic	Serine protease activity

Dot plots: ejemplo



Dot plots: ejemplo (cont.)

Dominios repetidos muestran un patrón característico.



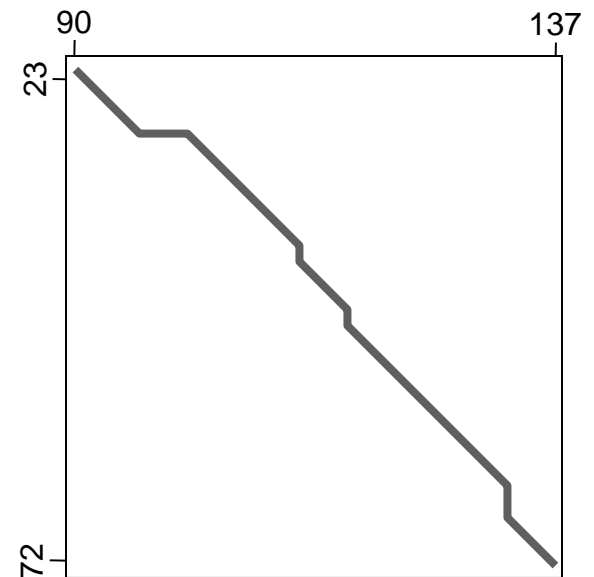
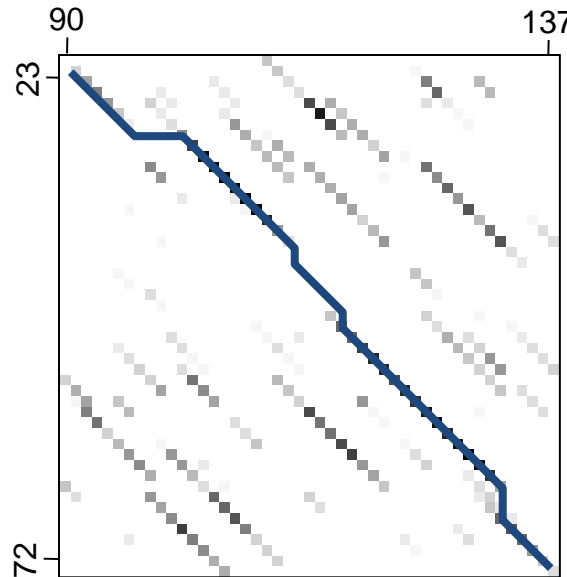
Dot plots: path graphs

Dot plots sugieren caminos (paths) a través del espacio de alineamientos posibles.

Path graphs son representaciones más explícitas de un alineamiento.

Cada path es un alineamiento único.

Dominios EGF conservados en la urokinase plasminogen activator (PLAU) y el tissue plasminogen activator (PLAT)



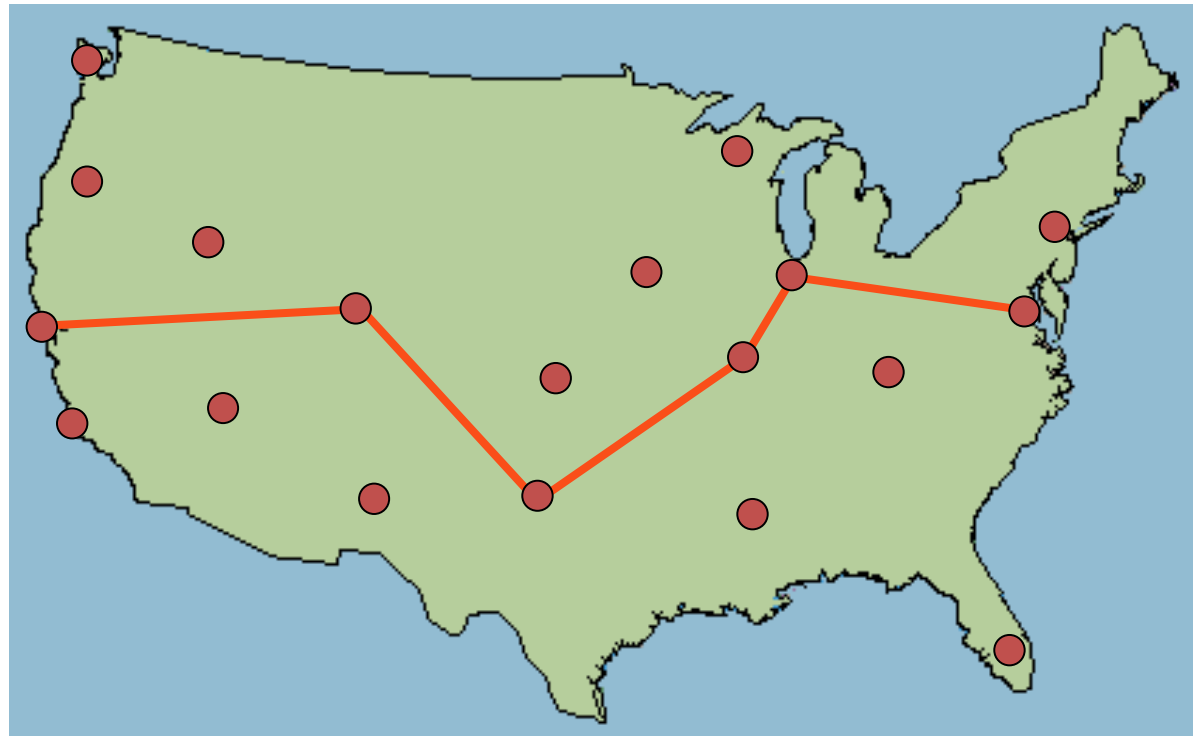
PLAU	90	EPKKVKDHC	SKHSPCQKGGTCVNMP--SGPH-CLCPQH	LTGNHCQKEK---CFE	137
PLAT	23	ELHQVPSN	CD----CLNGGTCVSNKYFSNIHWCNCPKKF	GGQHCEIDKSKTCYE	72

Path graphs: encontrar el mejor camino

Los problemas que involucran encontrar la mejor ruta o camino (Best-path problems) son comunes en computación científica.

El algoritmo para encontrar el mejor camino entre dos extremos y pasando por varios puntos se llama 'dynamic programming'

Rutear una llamada telefónica desde NY a San Francisco



Dynamic programming: introducción

Un ejemplo:

Construir un
alineamiento óptimo
entre estas dos
secuencias

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

Utilizando las
siguientes reglas de
scoring:

Match: +1

Mismatch: -1

Gap: -1

Dynamic programming: ejemplo

Ordenar las dos
secuencias en una
matriz bidimensional

Los vértices de cada
celda se encuentran
entre letras (bases).

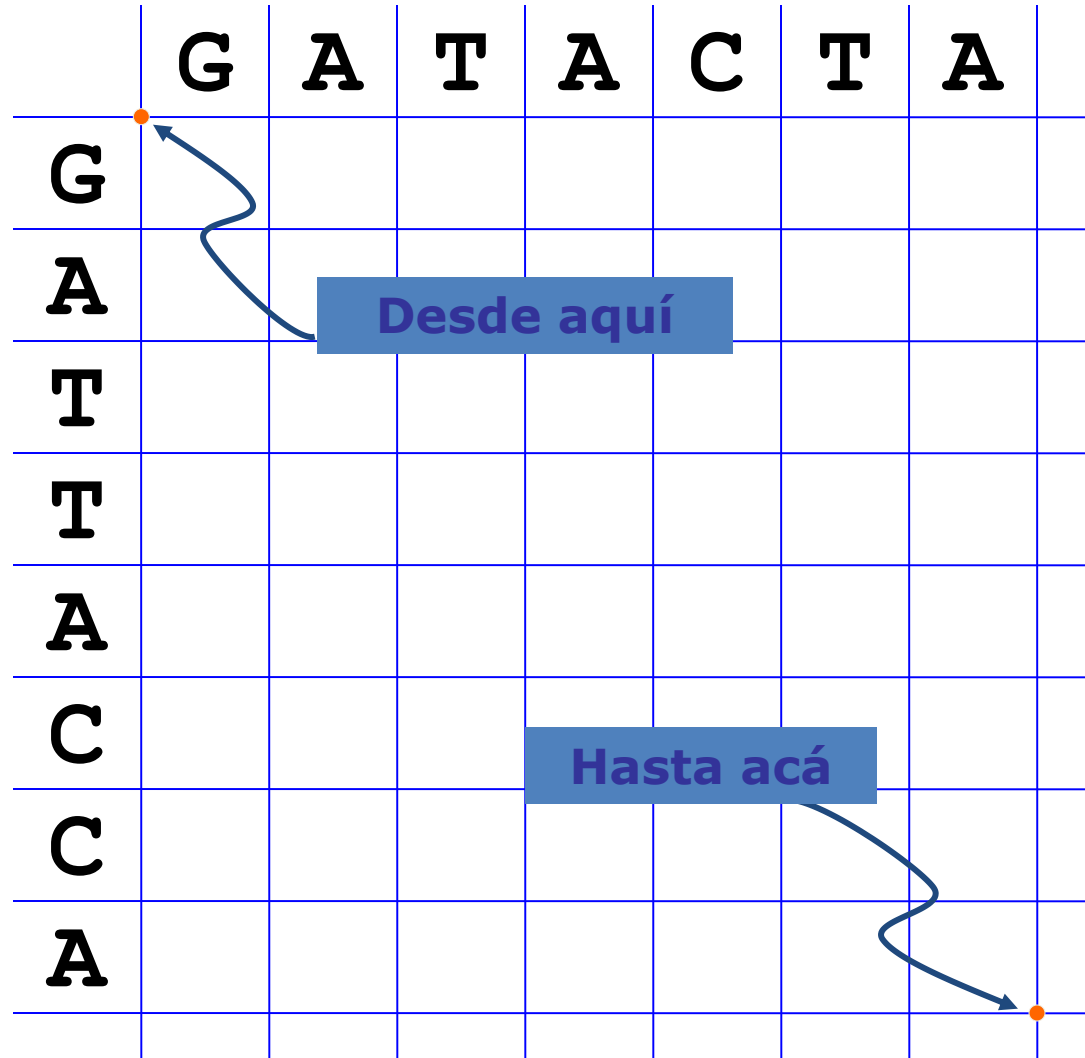
Needleman & Wunsch
(1970)

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

Slides Dynamic Programming: Hugues Sicotte (NCBI)

Dynamic programming: ejemplo (cont.)

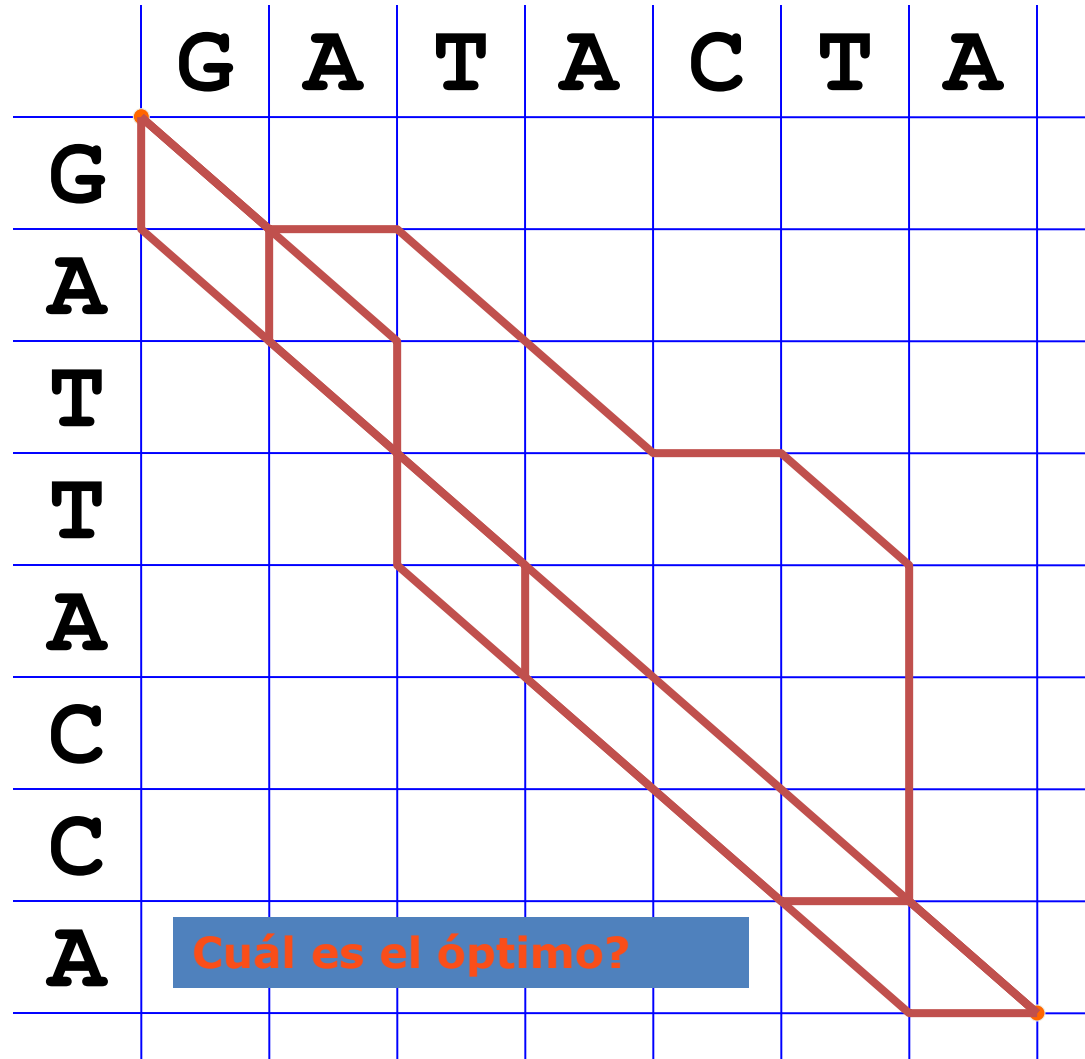
El objetivo es
encontrar la ruta
(path) óptimo



Slides Dynamic Programming: Hugues Sicotte (NCBI)

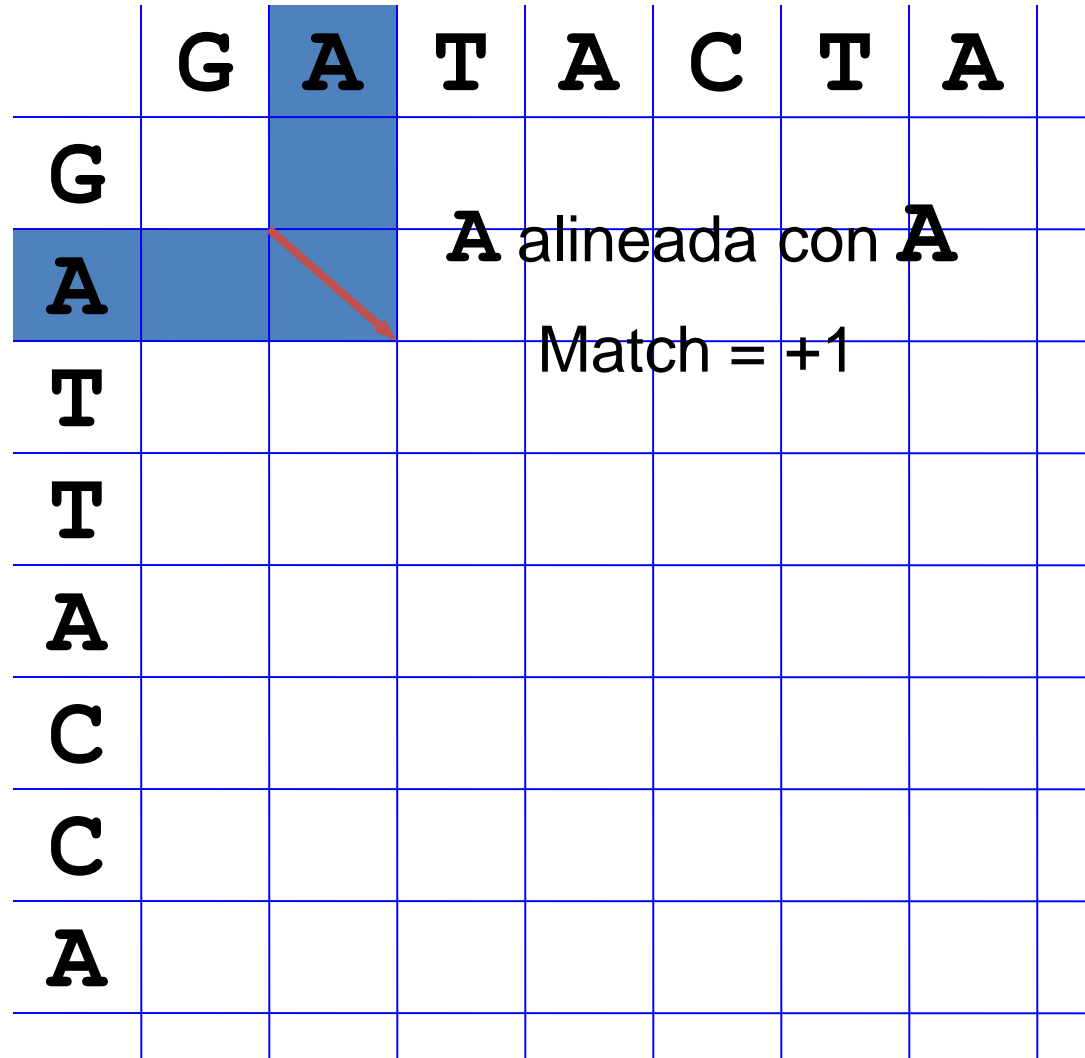
Dynamic programming: paths posibles

Cada path corresponde a
un alineamiento único



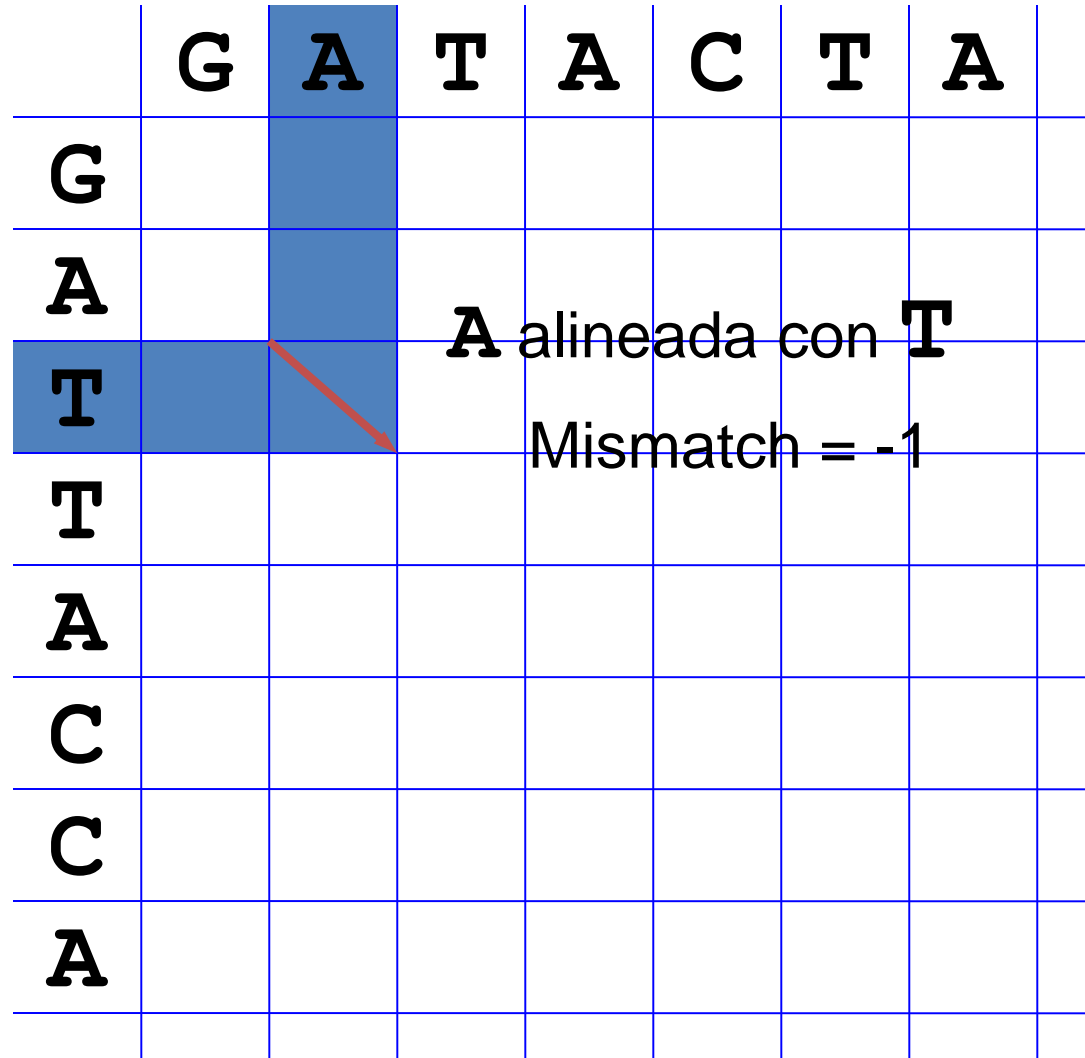
Dynamic programming: scores: match

El score para una ruta (path) es la suma incremental de los scores de sus pasos (diagonales o lados).



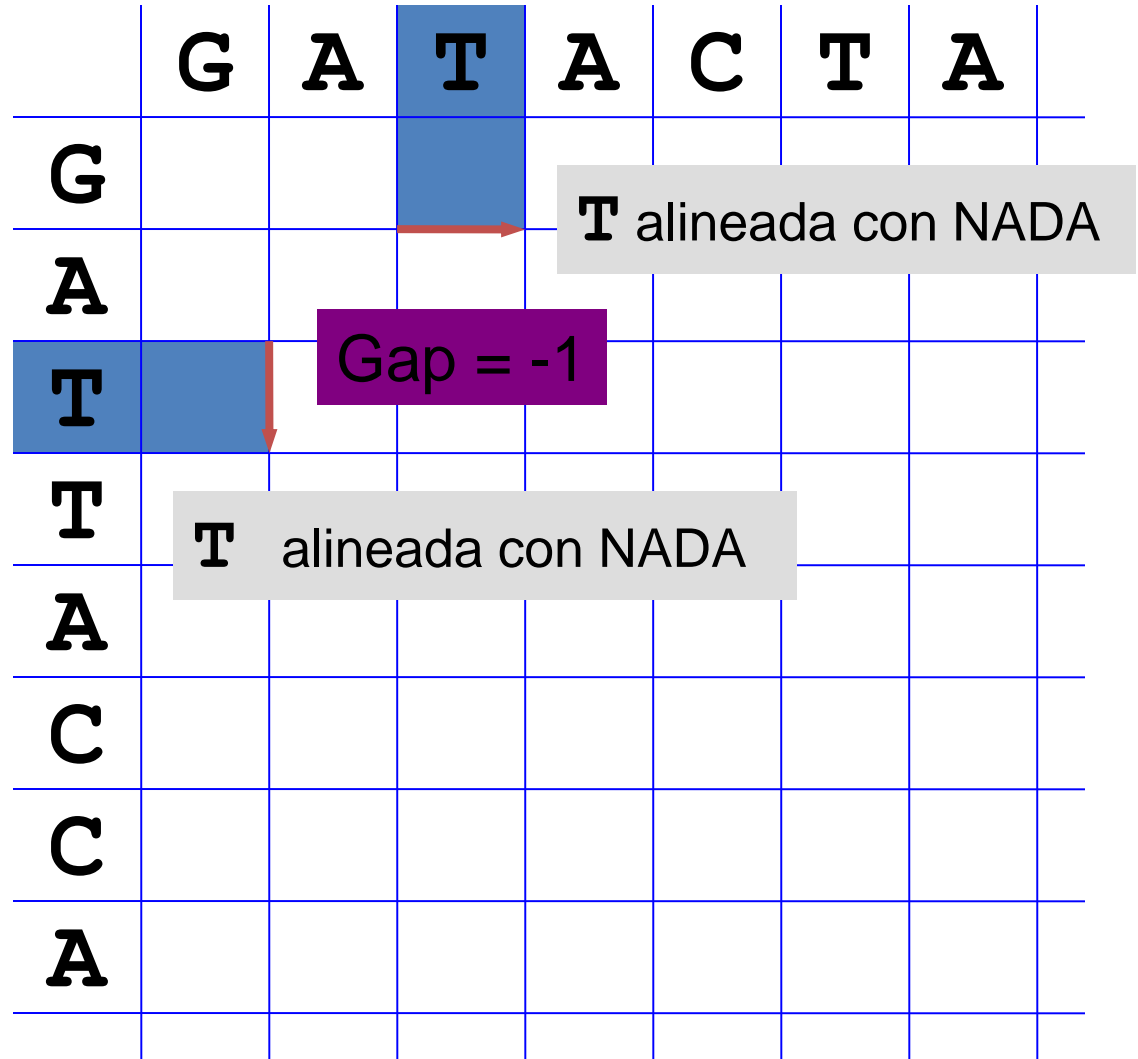
Dynamic programming: scores: mismatch

El score para una ruta (path) es la suma incremental de los scores de sus pasos (diagonales o lados).



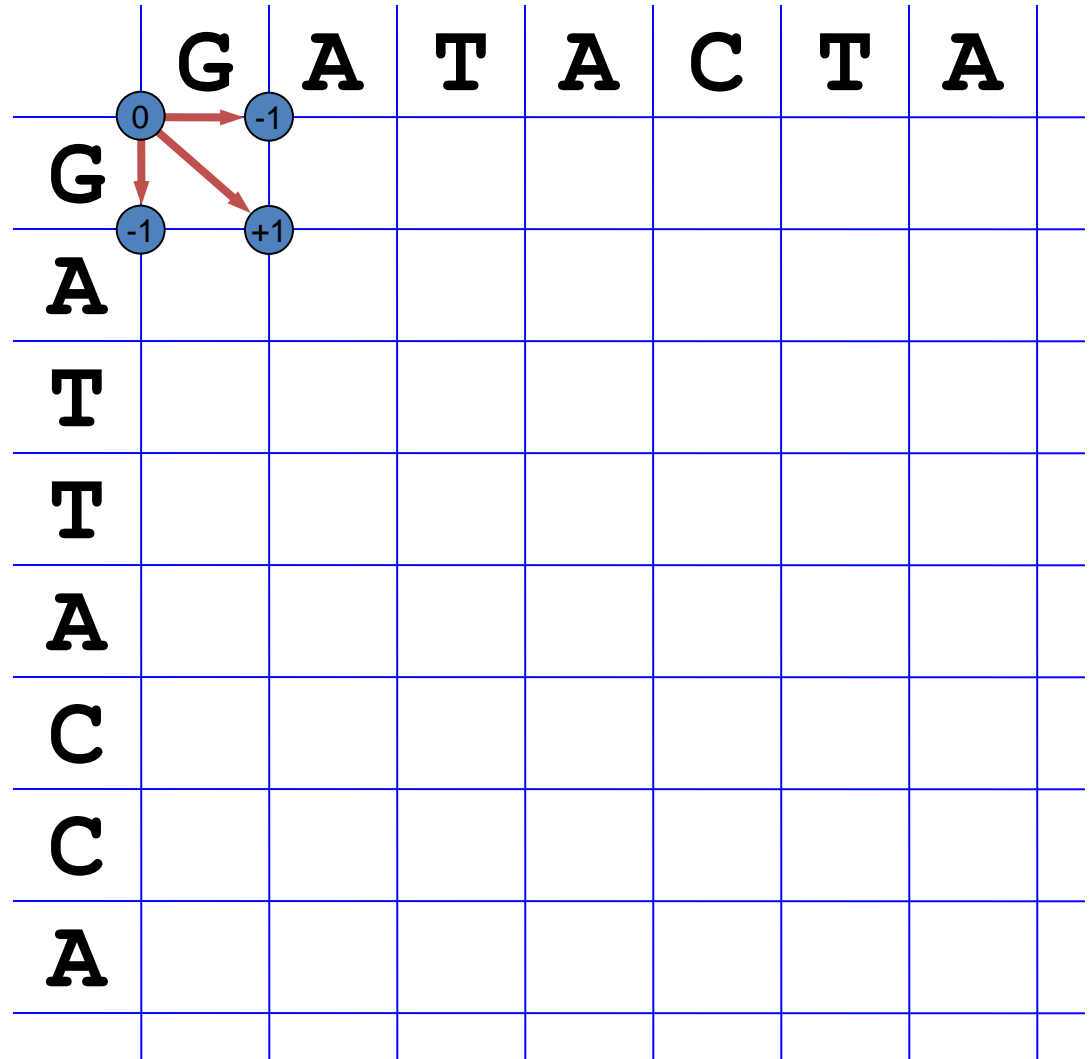
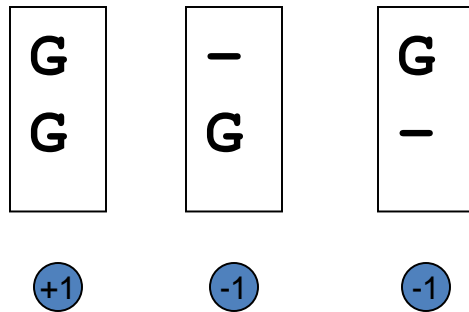
Dynamic programming: scores: gaps

El score para una ruta (path) es la suma incremental de los scores de sus pasos (diagonales o lados).



Dynamic programming: paso a paso (1)

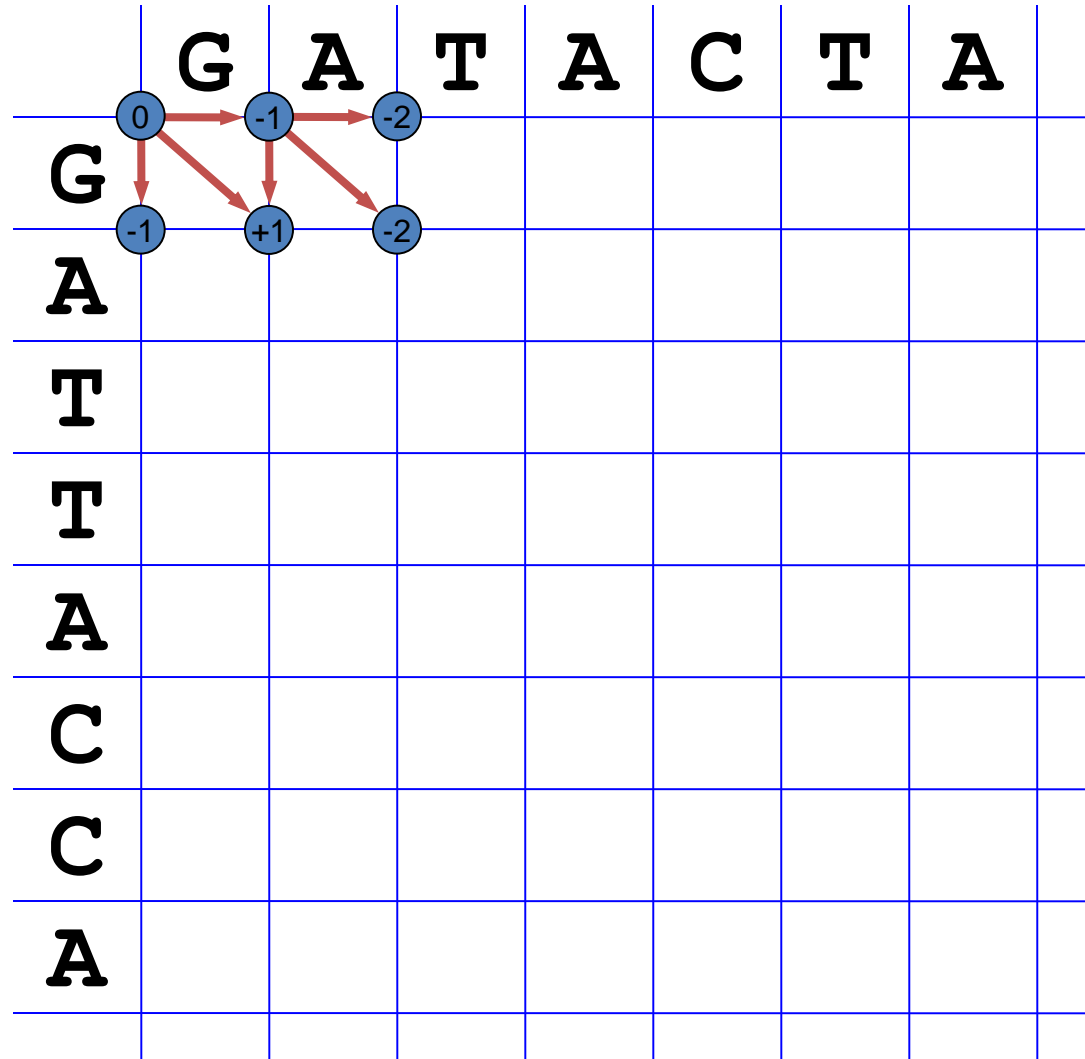
Extender el path paso por paso



Dynamic programming: paso a paso (2)

Incrementar el path paso a paso

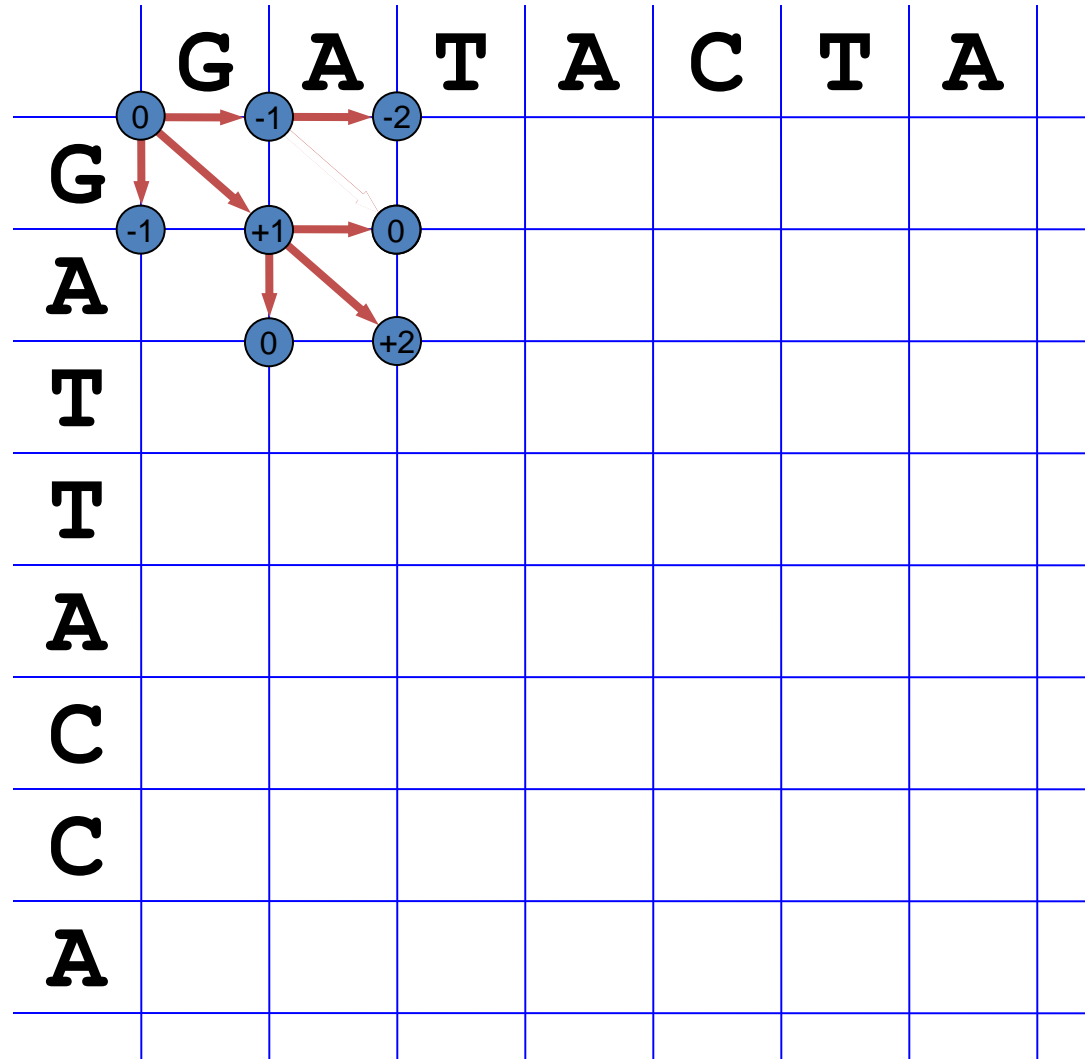
Recordar el mejor subpath que lleva a cada punto en la matriz.



Dynamic programming: paso a paso (3)

Incrementar el path paso a paso

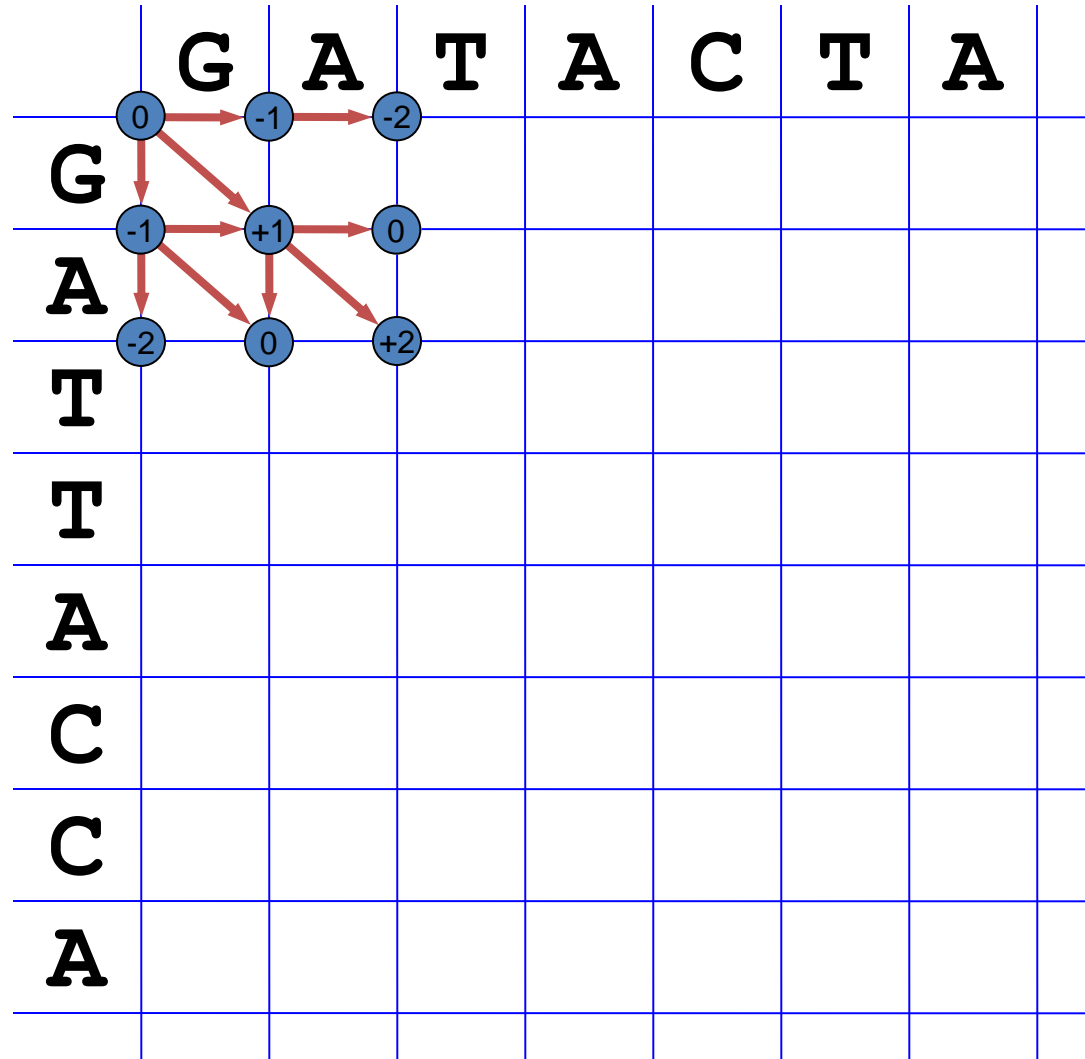
Recordar el mejor subpath que lleva a cada punto en la matriz.



Dynamic programming: paso a paso (4)

Incrementar el path paso a paso

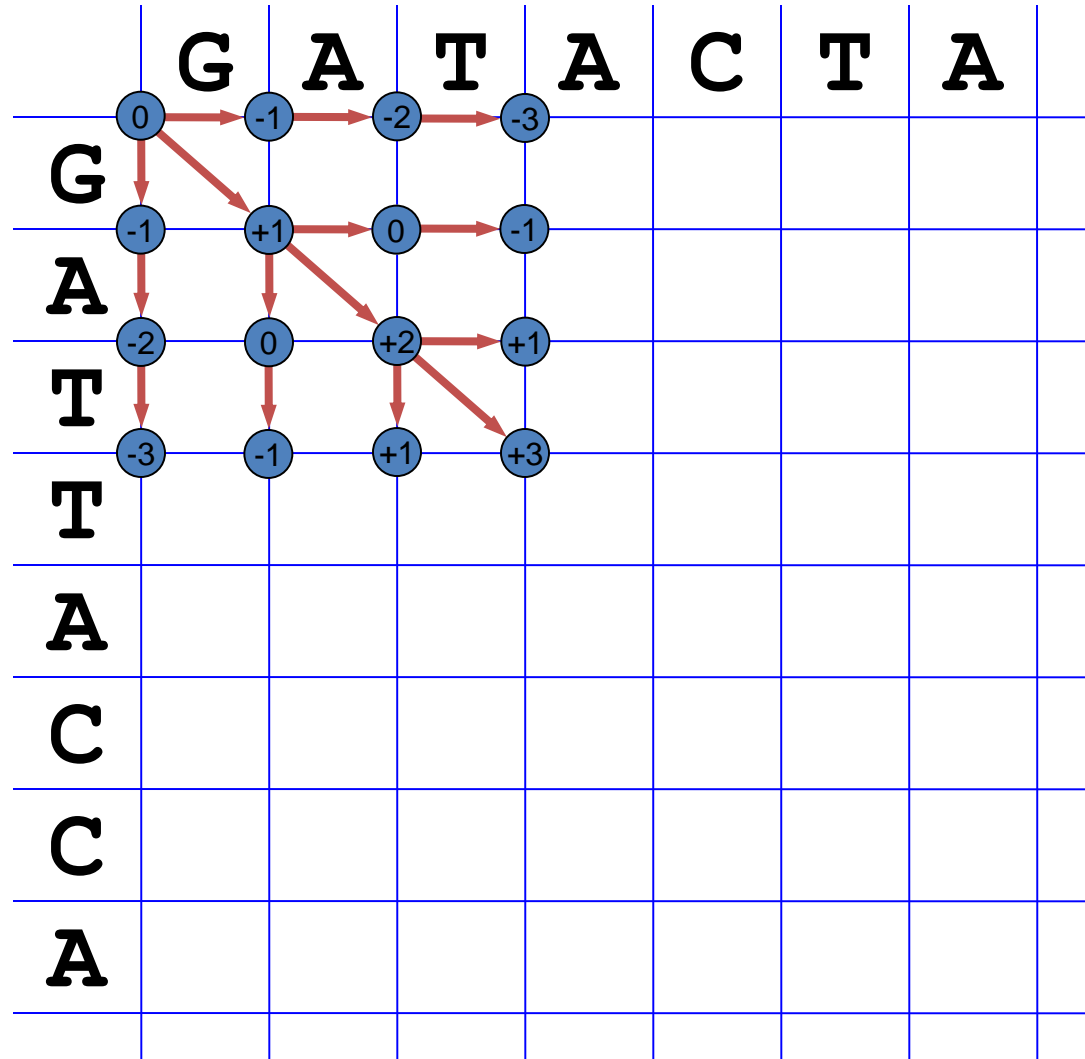
Recordar el mejor subpath que lleva a cada punto en la matriz.



Dynamic programming: paso a paso (5)

Incrementar el path paso a paso

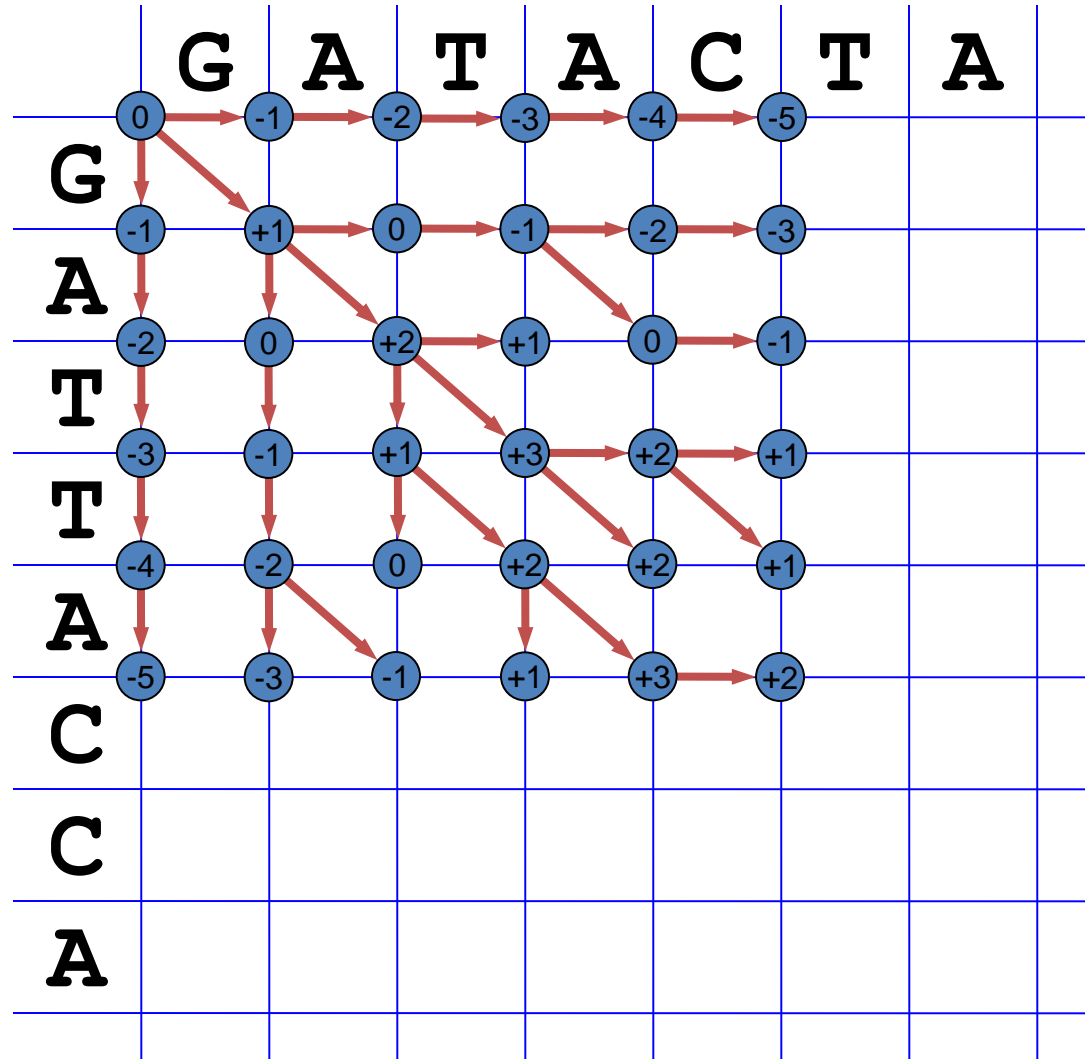
Recordar el mejor subpath que lleva a cada punto en la matriz.



Dynamic programming: paso a paso (6)

Incrementar el path paso a paso

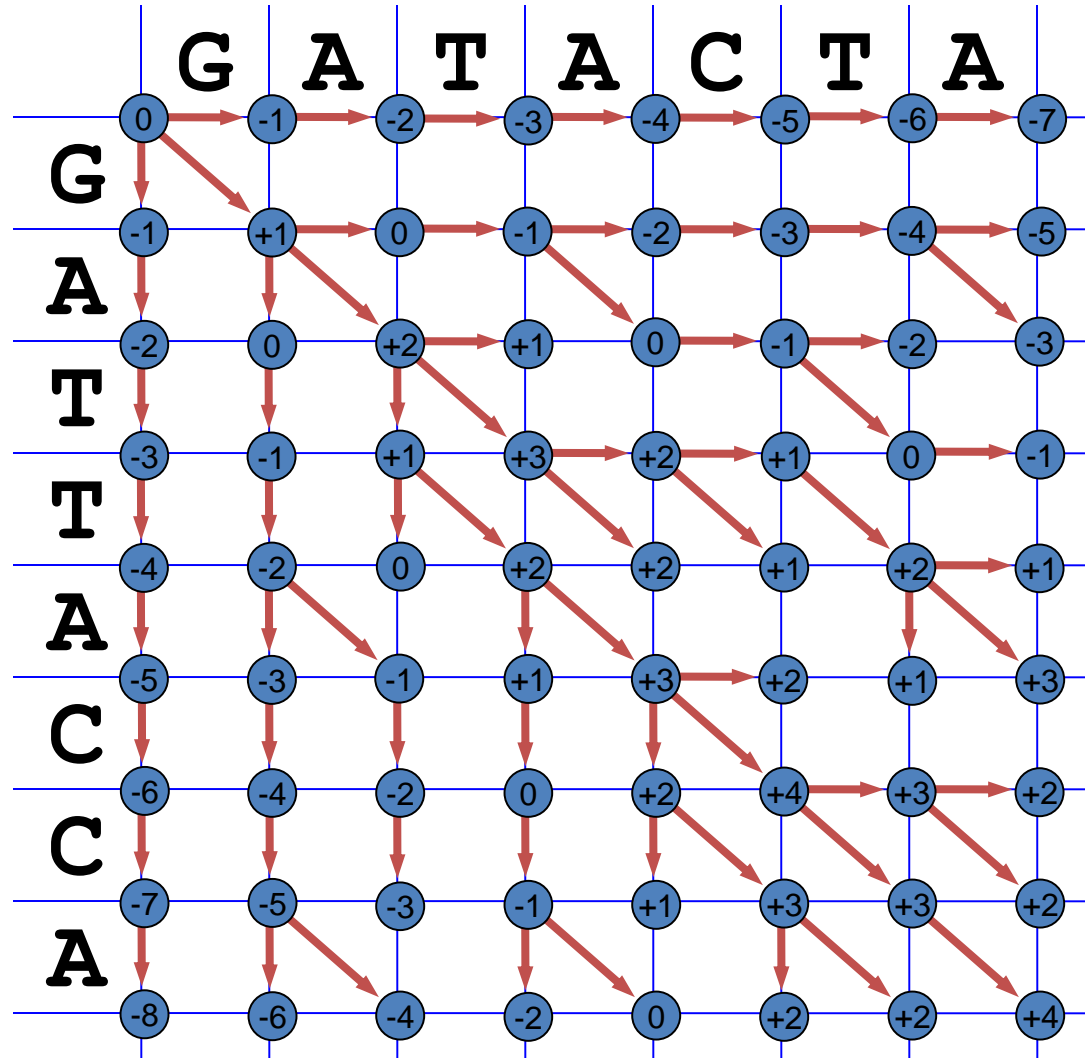
Recordar el mejor subpath que lleva a cada punto en la matriz.



Dynamic programming: paso a paso (7)

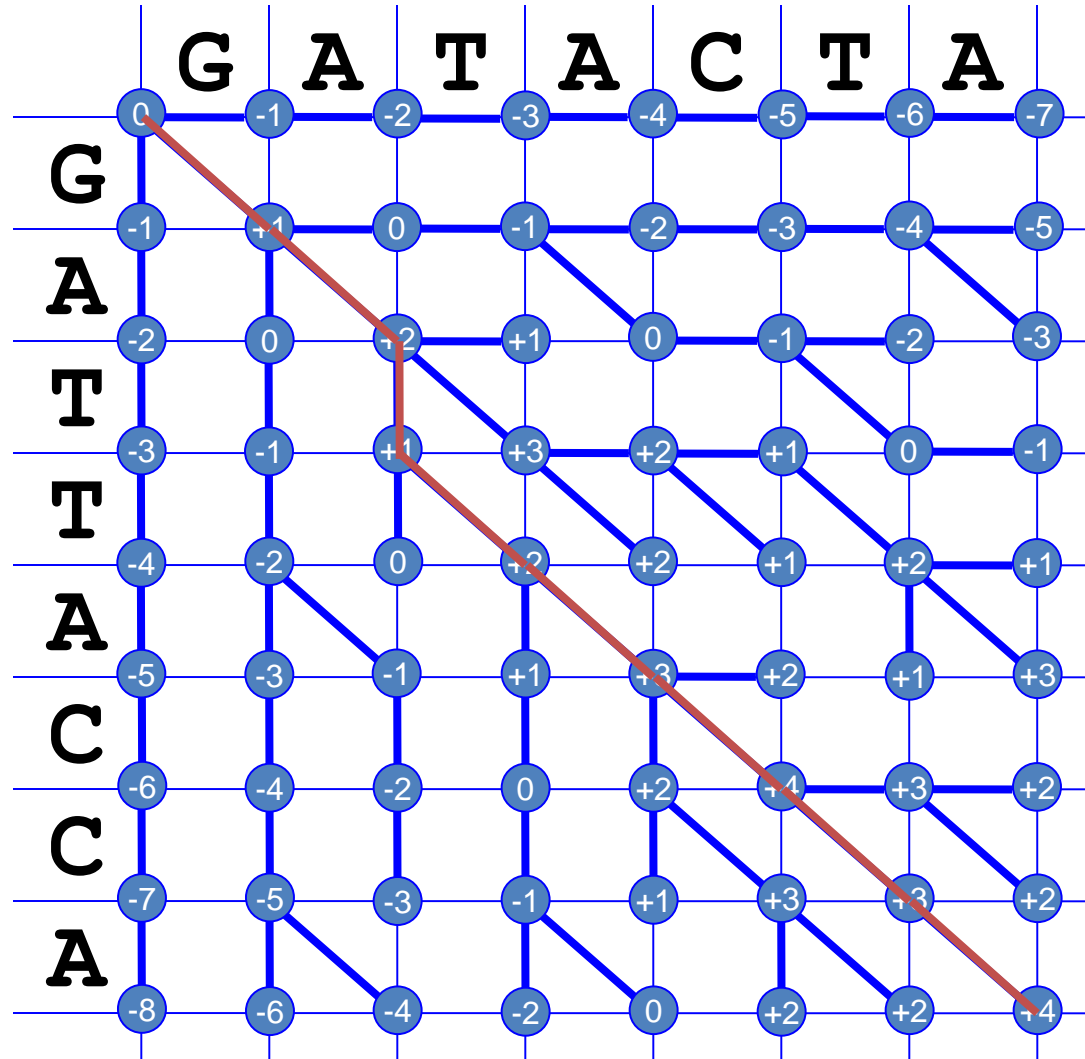
Incrementar el path paso a paso

Recordar el mejor subpath que lleva a cada punto en la matriz.



Dynamic programming: best path

Recorrer el camino de atrás hacia adelante para obtener el mejor path y alineamiento.

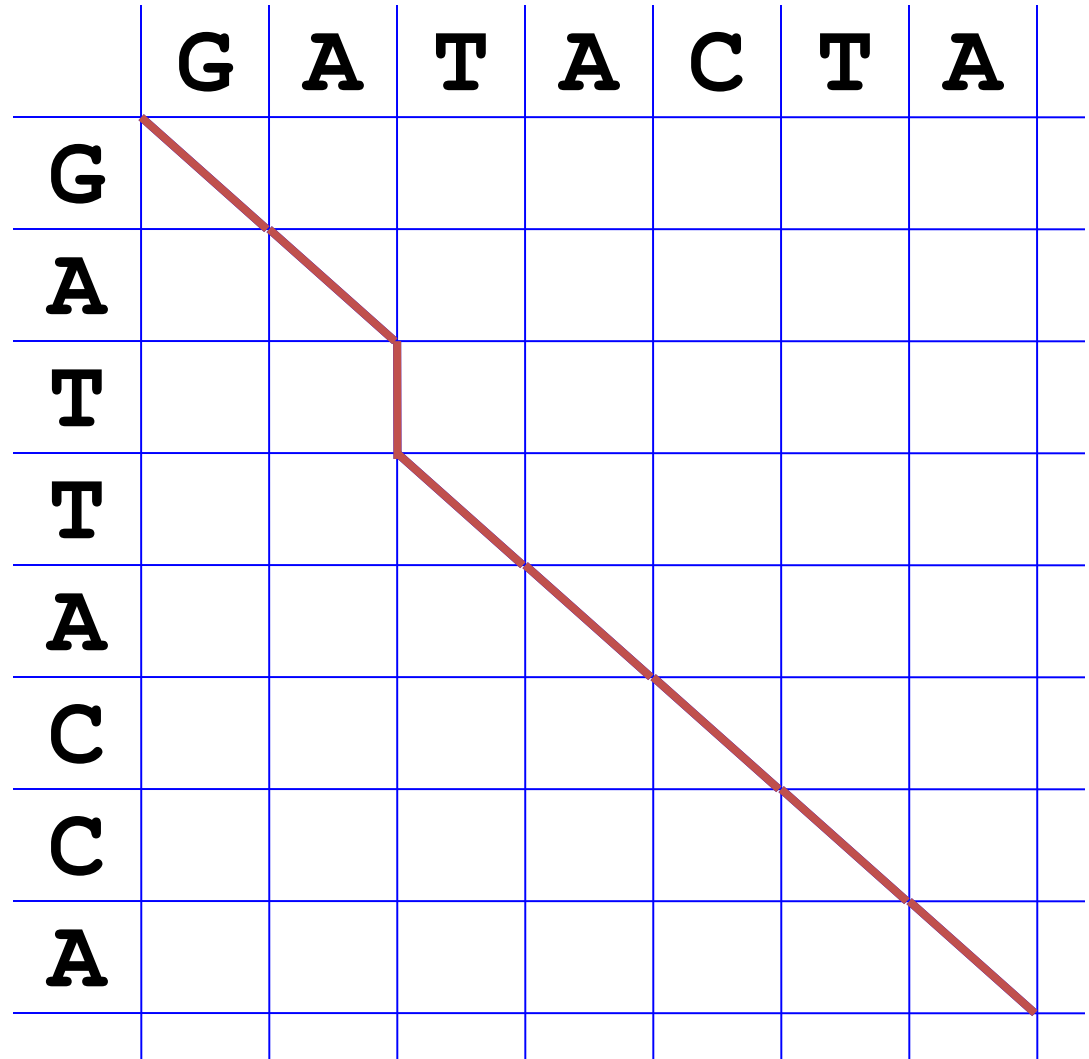


Slides Dynamic Programming: Hugues Sicotte (NCBI)

Dynamic programming: alineamiento obtenido

Imprimir el alineamiento

GA-TACTA
GATTACCA



Slides Dynamic Programming: Hugues Sicotte (NCBI)

Dynamic programming: Smith-Waterman

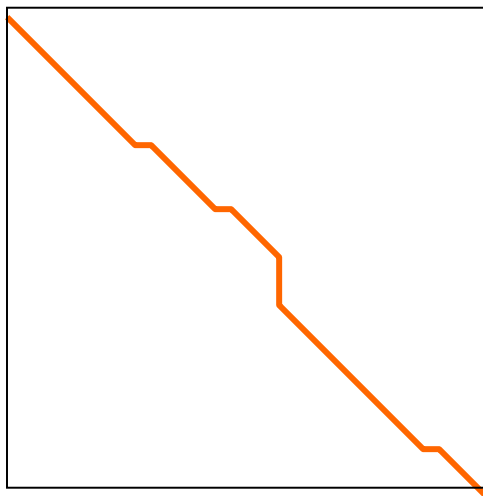
- El método fue modificado (Smith-Waterman) para obtener alineamientos locales
- El método garantiza la obtención de un alineamiento óptimo (cuyo score no puede ser mejorado)
- La complejidad es proporcional al producto de las longitudes de las secuencias a alinear

Similitud global y local

El algoritmo de programación dinámica puede ser implementado para alineamientos locales o globales.

Optimal global alignment

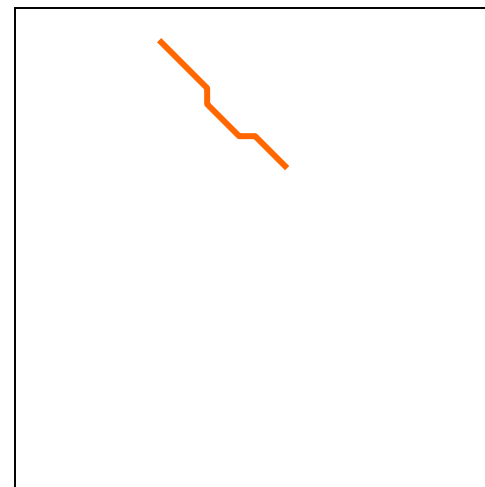
Needleman & Wunsch (1970)



Las secuencias se alinean esencialmente de un extremo a otro

Optimal local alignment

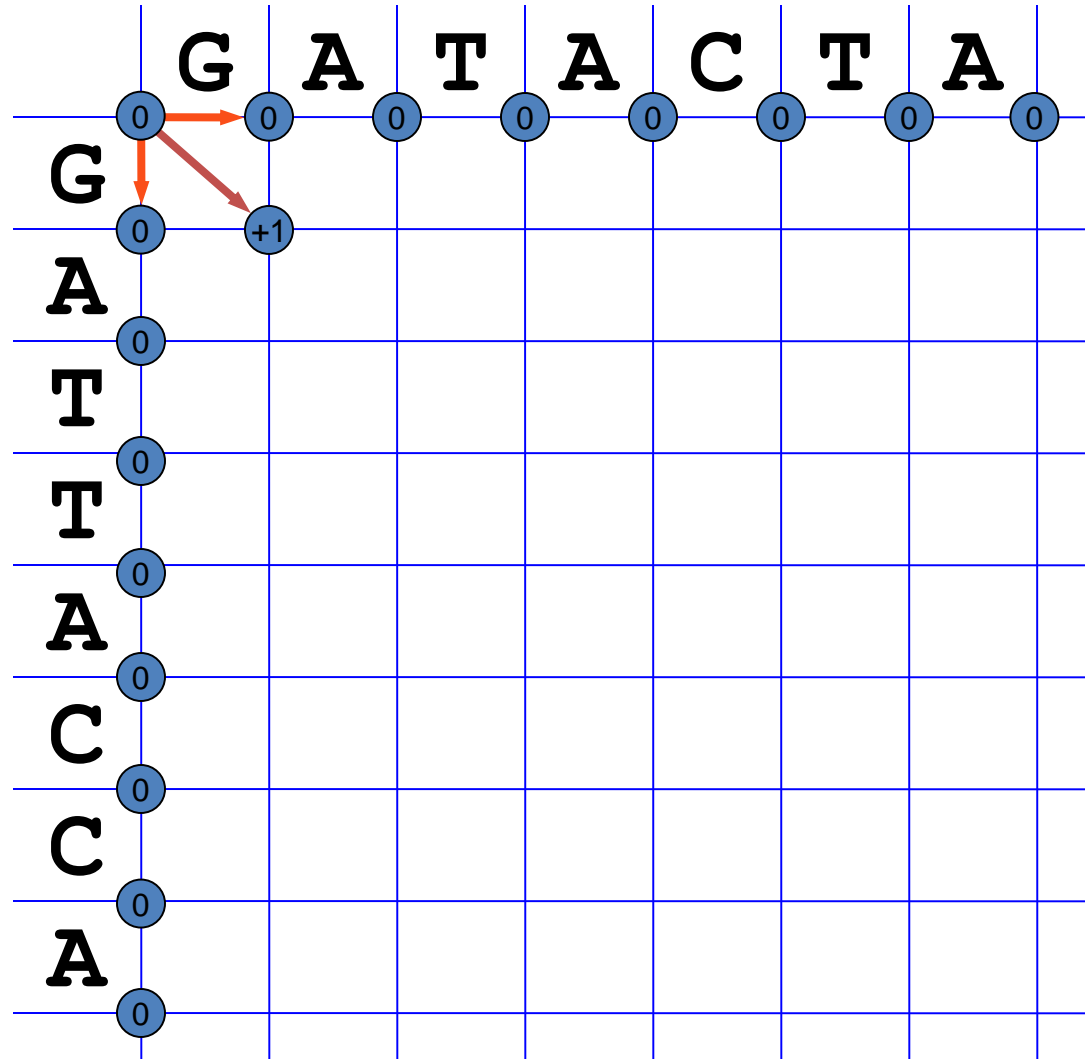
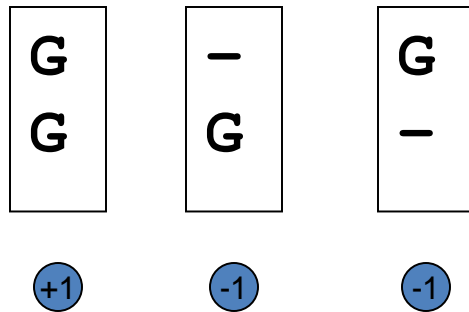
Smith & Waterman (1981)



Las secuencias se alinean en regiones pequeñas y aisladas

Smith-Waterman: paso a paso

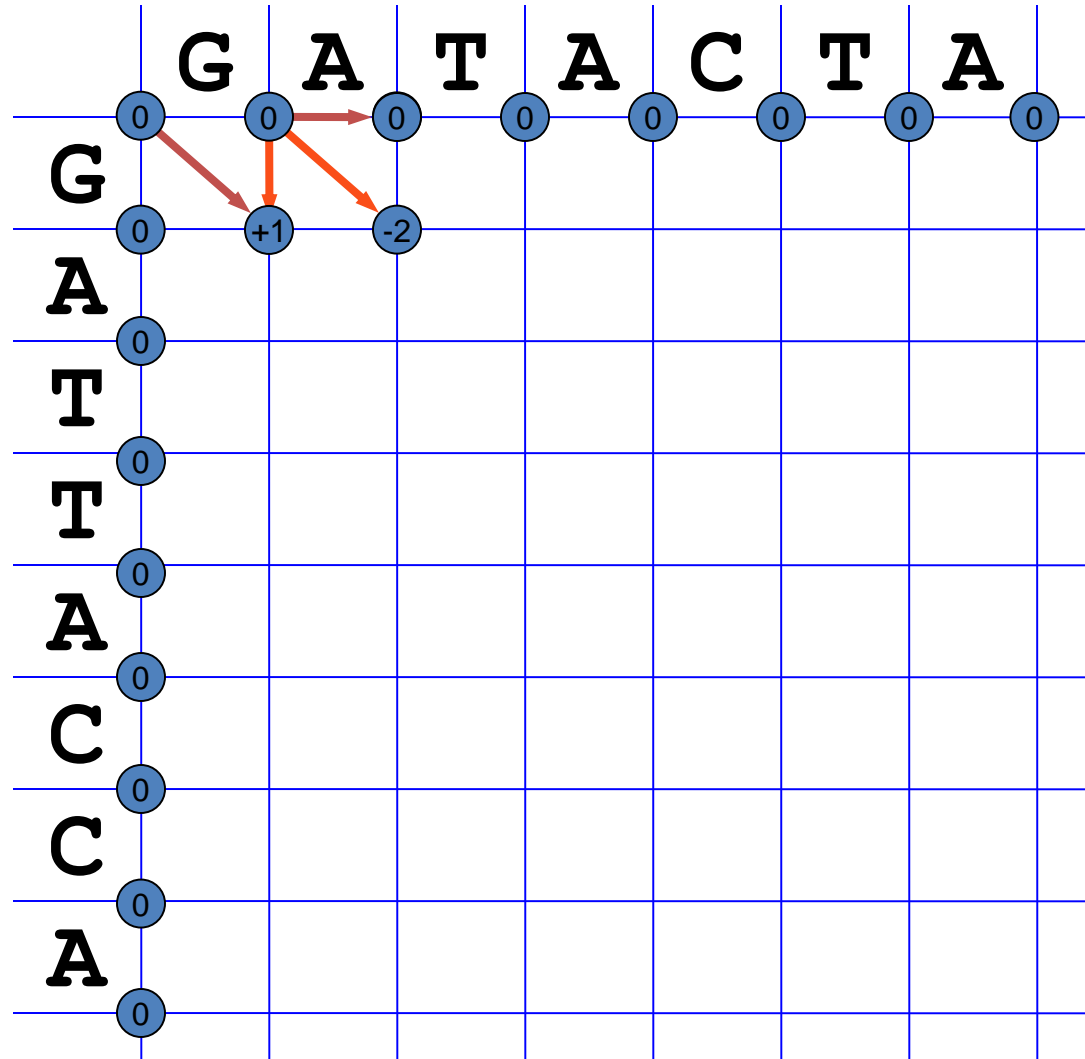
Extender el path paso por paso



Smith-Waterman: paso a paso (2)

Incrementar el path paso a paso

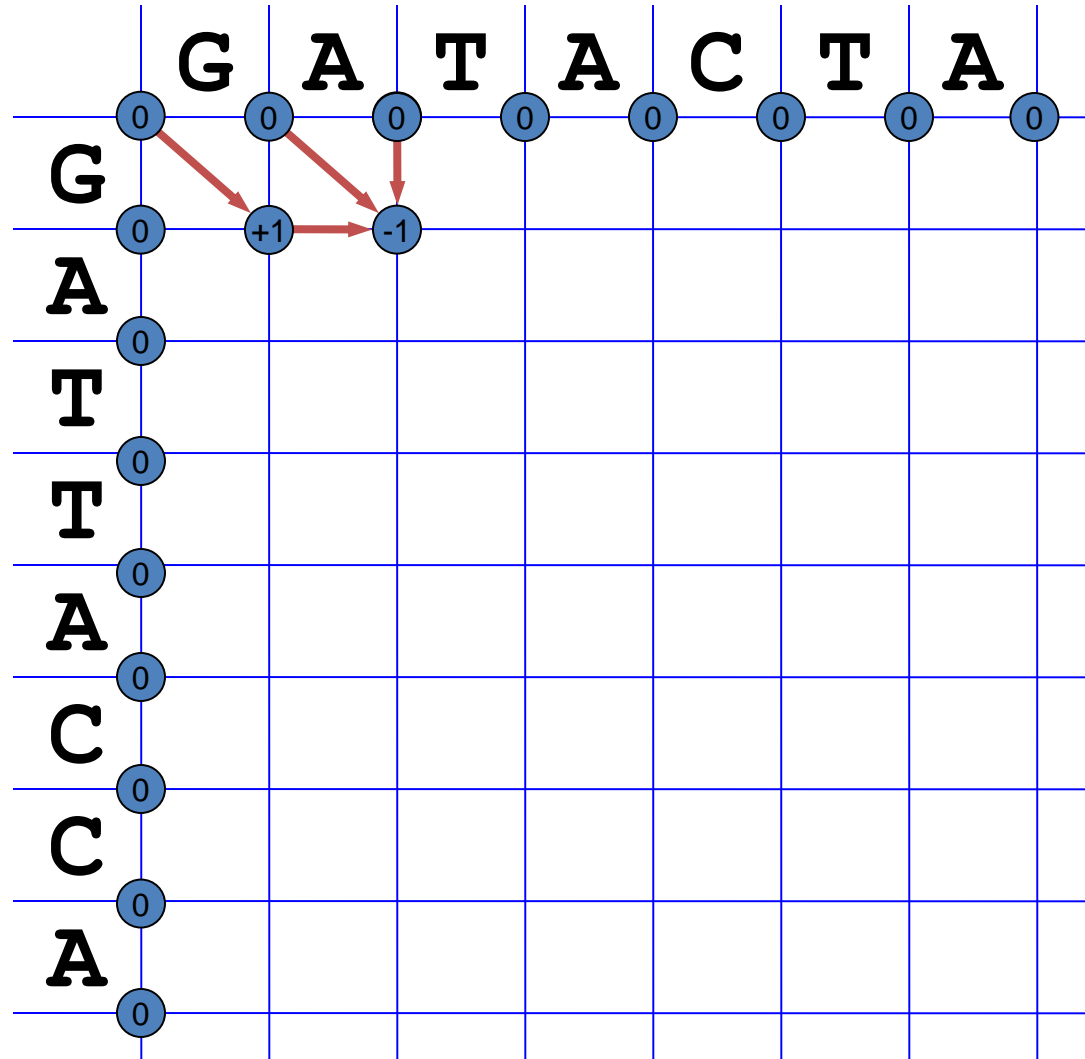
Recordar el mejor subpath que lleva a cada punto en la matriz.



Smith-Waterman: paso a paso (3)

Incrementar el path paso a paso

Recordar el mejor subpath que lleva a cada punto en la matriz.



Dynamic programming demos

- Needleman-Wunsch web app desarrollada para enseñanza

- <http://rna.informatik.uni-freiburg.de/Teaching/index.jsp?toolName=Needleman-Wunsch>

Input:

Sequence *a*:

Sequence *b*:

Optimization of: Distance ☐ Similarity ☒

Scoring in *s*: Match Mismatch Gap

Hint:
For similarity maximization,
match scores should be positive and all other scores lower.
For distance minimization the reverse applies.

Recursion:
$$D_{i,j} = \max \begin{cases} D_{i-1,j-1} + s(a_i, b_j) \\ D_{i-1,j} + s(a_i, -) \\ D_{i,j-1} + s(-, b_j) \end{cases} = \max \begin{cases} D_{i-1,j-1} + 1 & a_i = b_j \\ D_{i-1,j-1} + -1 & a_i \neq b_j \\ D_{i-1,j} + -2 & b_j = - \\ D_{i,j-1} + -2 & a_i = - \end{cases}$$

Output:

<i>D</i>		A ₁	A ₂	C ₃	G ₄
	0	-2	-4	-6	-8
A ₁	-2	1	-1	-3	-5
A ₂	-4	-1	2	0	-2
T ₃	-6	-3	0	1	-1
C ₄	-8	-5	-2	1	0
G ₅	-10	-7	-4	-1	2

Score: 2

Results
You can select a result to get the related traceback.

AATCG

** **

AA CG

Dynamic programming demos

- Smith-Waterman web app desarrollada para enseñanza
 - <http://rna.informatik.uni-freiburg.de/Teaching/index.jsp?toolName=Smith-Waterman>

Input:

Sequence a :

Sequence b :

Scoring in s : Match Mismatch Gap

Hint:
For similarity maximization,
match scores should be positive and all other scores lower.

Recursion:
$$S_{i,j} = \max \begin{cases} S_{i-1,j-1} + s(a_i, b_j) \\ S_{i-1,j} + s(a_i, -) \\ S_{i,j-1} + s(-, b_j) \\ 0 \end{cases} = \max \begin{cases} S_{i-1,j-1} + 1 & a_i = b_j \\ S_{i-1,j-1} + -1 & a_i \neq b_j \\ S_{i-1,j} + -2 & b_j = - \\ S_{i,j-1} + -2 & a_i = - \\ 0 \end{cases}$$

Output:

S		A_1	A_2	C_3	G_4
		0	0	0	0
A_1	0	1	1	0	0
A_2	0	1	2	0	0
T_3	0	0	0	1	0
C_4	0	0	0	1	0
G_5	0	0	0	0	2

Score: 2

Results
You can select a result to get the related traceback.

AA

**

AA

CG

**

CG

Global y local

- Un algoritmo de alineamiento local, siempre produce alineamientos locales?
- Un algoritmo de alineamiento global siempre produce alineamientos globales?
- **NO**
 - dependiendo del sistema de scoring (scores para match/mismatch/gaps) SW puede producir alineamientos globales
 - dependiendo la penalidad asignada a los gaps en los extremos de un alineamiento global (o alterando significativamente el sistema de scoring) NW puede producir alineamientos locales

Matrices

- **Un sistema de scoring simple, penaliza por igual cualquier mismatch**
- **Biológicamente tiene sentido penalizar ciertos cambios y ser más permisivo con otros**
 - **En proteínas: residuos hidrofóbicos reemplazados entre sí.**
 - **En DNA: transversiones vs transiciones**
- **Una matriz no es otra cosa que un sistema de scoring que permite asignar puntajes individuales a cada una de las letras del alfabeto en uso.**

- **Un ejemplo de matriz de scoring podría ser el clásico ejemplo de penalizar más los cambios que alteran las propiedades químicas de un residuo (aa)**
 - **hidrofóbicos: Ile, Val, Leu, Ala**
 - **Polares (+): Lys, Arg**
 - **Polares (-): Glu, Asp**
 - **Aromáticos: Phe, Tyr, Trp**
 - **etc.**

Ile x Val = -1

Ile x Asp = -5

Phe x Tyr = -1

Phe x Gly = -8

Matrices derivadas por observación

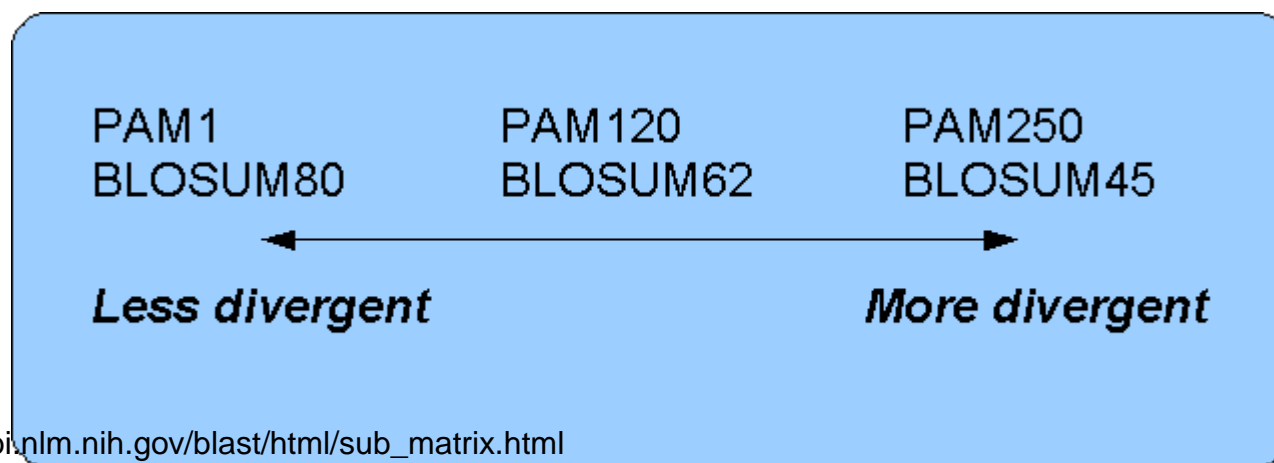
- **PAM (Dayhoff, 1978)**

- **proveen estimaciones de plausibilidad de cambio de un aminoácido en otro en proteínas homólogas**
- **derivadas a partir de un grupo de secuencias > 85% similares**
- **los cambios de aminoácidos observados son llamados “accepted mutations”**
- **Se extrapolan matrices a períodos evolutivos más largos**

Matrices derivadas por observación

- **BLOSUM (Henikoff)**

- **Blocks Amino Acid Substitution Matrices**
- **Sustituciones de amino ácidos observadas en un conjunto grande de 'blocks'**
- **Representan más de 500 familias de proteínas**
- **Se agrupan los blocks de acuerdo a su identidad y se generan matrices**
- **blocks 80% idénticos -> BLOSUM80**
- **Blocks 60% idénticos -> BLOSUM60**
- **etc**



https://www.ncbi.nlm.nih.gov/blast/html/sub_matrix.html

Sistemas de scoring: BLOSUM62

Algunas sustituciones son más comunes que otras

Los scores provienen de la observación de los tipos y frecuencias de sustitución en distintas familias proteicas

BLOSUM62

[illegible]

Sistemas de scoring: BLOSUM62: identidades

Las identidades tienen scores positivos, pero algunas son más valoradas que otras.

BLOSUM62

A	4																			
R	-1	5																		
N	-2	0	6																	
D	-2	-2	1	6																
C	0	-3	-3	-3	9															
Q	-1	1	0	0	-3	5														
E	-1	0	0	2	-4	2	5													
G	0	-2	0	-1	-3	-2	-2	6												
H	-2	0	1	-1	-3	0	0	-2	8											
I	-1	-3	-3	-3	-1	-3	-3	-4	-3	4										
L	-1	-2	-3	-4	-1	-2	-3	-4	-3	2	4									
K	-1	2	0	-1	-3	1	1	-2	-1	-3	-2	5								
M	-1	-1	-2	-3	-1	0	-2	-3	-2	1	2	-1	5							
F	-2	-3	-3	-3	-2	-3	-3	-3	-1	0	0	-3	0	6						
P	-1	-2	-2	-1	-3	-1	-1	-2	-2	-3	-3	-1	-2	-4	7					
S	1	-1	1	0	-1	0	0	0	-1	-2	-2	0	-1	-2	-1	4				
T	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-1	-1	-1	-2	-1	1	5			
W	-3	-3	-4	-4	-2	-2	-3	-2	-2	-3	-2	-3	-1	1	-4	-3	-2	11		
Y	-2	-2	-2	-3	-2	-1	-2	-3	2	-1	-1	-2	-1	3	-3	-2	-2	2	7	
V	0	-3	-3	-3	-1	-2	-2	-3	-3	3	1	-2	1	-1	-2	-2	0	-3	-1	4
	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V

Sistemas de scoring: BLOSUM62: sustituciones

Algunas sustituciones tienen scores positivos, pero la mayoría son negativos.

BLOSUM62

[illegible]

Más matrices

- **PAM**
- **BLOSUM**
- **Otras**
 - **Comparación simple de propiedades químicas de amino ácidos**
 - **Análisis complejos de sustituciones en estructura secundaria de proteínas, a partir de alineamientos estructurales**
 - **Gonnet (1994). Sustitución de dipéptidos**
 - **Jones (1994) matriz específica de proteínas transmembrana**
- **Algunas de estas matrices sirven para alinear proteínas en base a características estructurales y pueden no ser útiles para análisis evolutivos!**

Búsqueda de secuencias por similitud

- **Tenemos un método (algoritmo) que nos garantiza un alineamiento óptimo entre dos secuencias**
- **Tenemos un sistema de scoring complejo que refleja mejor nuestras ideas biológicas acerca de lo que es un alineamiento**
- **Cómo usaríamos estas herramientas para implementar una búsqueda por similitud contra una base de datos?**

Usemos la fuerza bruta

- Tenemos una base de datos con secuencias
- Tenemos una secuencia 'query' en la que estamos interesados
- Podemos encontrar secuencias similares al query en la base de datos?
- Tomar una por una las secuencias de la base de datos
- Calcular un alineamiento y su score
- Elegir los mejores alineamientos en base al score
- Finalmente usar nuestro criterio y evaluar si la/s secuencia/s encontradas son lo suficientemente similares

Heurísticas para reducir espacio de búsqueda

- **Hay dos espacios de búsqueda reconocibles:**
 - El espacio de todas las secuencias de la base de datos
 - El espacio de todos los alineamientos posibles entre dos secuencias
- **Las búsquedas de secuencias por similitud son “exactas” si recorren completamente ambos espacios**
 - Ej: Smith-Waterman sobre toda la base de datos
- **A continuación vamos a introducir heurísticas para reducir estos espacios de búsqueda**
 - Estrategias de hashing para filtrar la base de datos
 - Distintas heurísticas para reducir el espacio de alineamientos posibles que se explora efectivamente

Búsquedas en bases de datos

**Compara una
secuencia (query)
contra una base de
datos de secuencias**

Una búsqueda
típica tiene 4
elementos básicos.

```
> fasta myquery swissprot -ktup 2
```

↑
Programa

↑
query

↑
Base de
datos

↑
Parámetros
opcionales

Búsqueda en bases de datos

Con el crecimiento exponencial de las bases de datos las búsquedas son cada vez más lentas ...

```
> fasta myquery swissprot -ktup 2  
  
searching .....
```

Database searching

La lista de hits provee los 'títulos' y scores de las secuencias que fueron seleccionadas por la secuencia 'query'.

> fasta myquery swissprot -ktup 2

```
The best scores are:
                                initn initl opt  z-sc E(77110)
gi|1706794|sp|P49789|FHIT_HUMAN BIS(5'-ADENOSYL)- 996 996 996 1262.1 0
gi|1703339|sp|P49776|APH1_SCHPO BIS(5'-NUCLEOSYL) 412 382 395 507.6 1.4e-21
gi|1723425|sp|P49775|HNT2_YEAST HIT FAMILY PROTEI 238 133 316 407.4 5.4e-16
gi|3915958|sp|Q58276|Y866_METJA HYPOTHETICAL HIT- 153 98 190 253.1 2.1e-07
gi|3916020|sp|Q11066|YHIT_MYCTU HYPOTHETICAL 15.7 163 163 184 244.8 6.1e-07
gi|3023940|sp|O07513|HIT_BACSU HIT PROTEIN 164 164 170 227.2 5.8e-06
gi|2506515|sp|Q04344|HNT1_YEAST HIT FAMILY PROTEI 130 91 157 210.3 5.1e-05
gi|2495235|sp|P75504|YHIT_MYCPN HYPOTHETICAL 16.1 125 125 148 199.7 0.0002
gi|418447|sp|P32084|YHIT_SYNP7 HYPOTHETICAL 12.4 42 42 140 191.3 0.00058
gi|3025190|sp|P94252|YHIT_BORBU HYPOTHETICAL 15.9 128 73 139 188.7 0.00082
gi|1351828|sp|P47378|YHIT_MYCGE HYPOTHETICAL HIT- 76 76 133 181.0 0.0022
gi|418446|sp|P32083|YHIT_MYCHR HYPOTHETICAL 13.1 27 27 119 165.2 0.017
gi|1708543|sp|P49773|IPK1_HUMAN HINT PROTEIN (PRO 66 66 118 163.0 0.022
gi|2495231|sp|P70349|IPK1_MOUSE HINT PROTEIN (PRO 65 65 116 160.5 0.03
gi|1724020|sp|P49774|YHIT_MYCLE HYPOTHETICAL HIT- 52 52 117 160.3 0.031
gi|1170581|sp|P16436|IPK1_BOVIN HINT PROTEIN (PRO 66 66 115 159.3 0.035
gi|2495232|sp|P80912|IPK1_RABIT HINT PROTEIN (PRO 66 66 112 155.5 0.057
gi|1177047|sp|P42856|ZB14_MAIZE 14 KD ZINC-BINDIN 73 73 112 155.4 0.058
gi|1177046|sp|P42855|ZB14_BRAJU 14 KD ZINC-BINDIN 76 76 110 153.8 0.072
gi|1169825|sp|P31764|GAL7_HAEIN GALACTOSE-1-PHOSP 58 58 104 138.5 0.51
gi|113999|sp|P16550|APA1_YEAST 5',5'''-P-1,P-4-TE 47 47 103 137.8 0.56
gi|1351948|sp|P49348|APA2_KLULA 5',5'''-P-1,P-4-T 63 63 98 131.3 1.3
gi|123331|sp|P23228|HMCS_CHICK HYDROXYMETHYLGLUTA 58 58 99 129.4 1.6
gi|1170899|sp|P06994|MDH_ECOLI MALATE DEHYDROGENA 70 48 91 122.9 3.7
gi|3915666|sp|Q10798|DXR_MYCTU 1-DEOXY-D-XYLULOSE 75 50 92 121.9 4.3
gi|124341|sp|P05113|IL5_HUMAN INTERLEUKIN-5 PRECU 36 36 85 121.3 4.7
gi|1170538|sp|P46685|IL5_CERTO INTERLEUKIN-5 PREC 36 36 84 120.0 5.5
gi|121369|sp|P15124|GLNA_METCA GLUTAMINE SYNTHETA 45 45 90 118.9 6.3
gi|2506868|sp|P33937|NAP_A_ECOLI PERIPLASMIC NITRA 48 48 92 117.4 7.6
gi|119377|sp|P10403|ENV1_DROME RETROVIRUS-RELATED 59 59 89 117.0 8
gi|1351041|sp|P48415|SC16_YEAST MULTIDOMAIN VESIC 48 48 97 117.0 8
gi|4033418|sp|O67501|IPYR_AQUAE INORGANIC PYROPHO 38 38 83 116.8 8.3
```

Búsquedas en bases de datos

El detalle de los
alineamientos se
muestra más abajo

```
> fasta myquery swissprot -ktup 2
```

```
>>gi|1703339|sp|P49776|APH1_SCHPO BIS(5'-NUCLEOSYL)-TETR (182 aa)
initn: 412  initl: 382  opt: 395 z-score: 507.6 E(): 1.4e-21
Smith-Waterman score: 395; 52.3% identity in 109 aa overlap
```

```

          10          20          30          40          50
gi|170  MSFRFGQHLIKPSVVFLKTELFSALVNRKPVVPGHVLVCPLRPVERFHDLRPDEVADLF
          : X: ..... : : : : : : : : : : : : : : : : : : : : : : : : :
gi|170  MPKQLYFSKFPVGSQVFYRTKLSAAFVNLKPILPGHVLVIPQRAVPRLKDLTPSELTDLF
          10          20          30          40          50          60
```

```

        60        70        80        90       100       110
gi|170  QTTQRVGTVVEKHFHGTSLTFSMQDGPEAGQTVKHVHVHVLPRKAGDFHRNDSIYEELQK
        ..... : : : : : : : : : : : : : : : : : : : : : : : : : : : : : : :
gi|170  TSVRKVQQVIEKVFSASASNIGIQDGVDAGQTVPHVHVHIIIPRKADFSENDLVYSELEK
          70        80        90       100       110       120
```

```

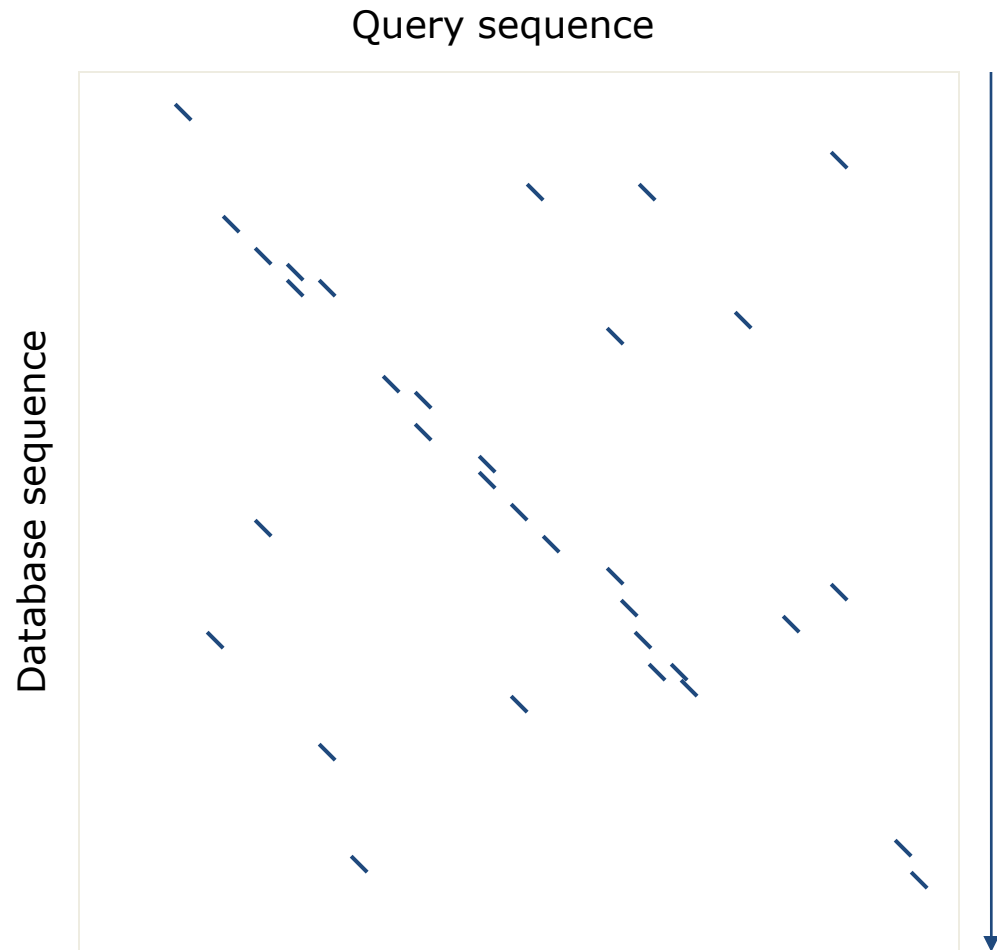
        120       130       140
gi|170  HDKEDFPASWRSEEEMAAEAAALRVYFQ
        ..
gi|170  NEGNLASLYLTGNERYAGDERPPTSMRQAIPKDEDRKPRTLEEMEKEAQWLKGYFSEEQE
          130       140       150       160       170       180
```

```
>>gi|1723425|sp|P49775|HNT2_YEAST HIT FAMILY PROTEIN 2 (217 aa)
initn: 238  initl: 133  opt: 316 z-score: 407.4 E(): 5.4e-16
Smith-Waterman score: 316; 37.4% identity in 131 aa overlap
```


Búsquedas en bases de datos: hashing methods

La búsqueda más simple es un gran ejemplo de dynamic programming. Para una secuencia query de **N** letras, contra una base de datos de **M** letras, se requieren **$M \times N$** comparaciones.

Cómo reducir este espacio de búsqueda?



Hashing methods

Hashing es un método común para acelerar búsquedas en bases de datos.

Compilar un “diccionario” de palabras a partir de la secuencia ‘query’. Armar un índice con todas las palabras.

Longitud de la secuencia:
19

Cantidad de palabras:
17

MLIIKRDELVISWASHERE

MLI
LII
IIK
IKR
KRD
RDE
DEL
ELV
LVI
VIS
ISW
SWA
WAS
ASH
SHE
HER
ERE

query
sequence

Todas las palabras
posibles de
longitud **ktup**

ktup = 3

Hashing methods

Construir el diccionario de palabras para la secuencia 'query' requiere $N-2$ operaciones.

La base de datos contiene $M-2$ palabras, con un límite máximo de 20^{ktup} palabras (proteínas = 20 aminoácidos o letras posibles)

Para $ktup=3$ el número total (máximo) de palabras es $20^3 = 8000$

Esta operación de búsqueda es muy eficiente computacionalmente

MLIIKRDELVISWASHERE

MLI

LII

IIK

IKR

KRD

RDE

DEL

ELV

LVI

VIS

ISW

SWA

WAS

ASH

SHE

HER

ERE

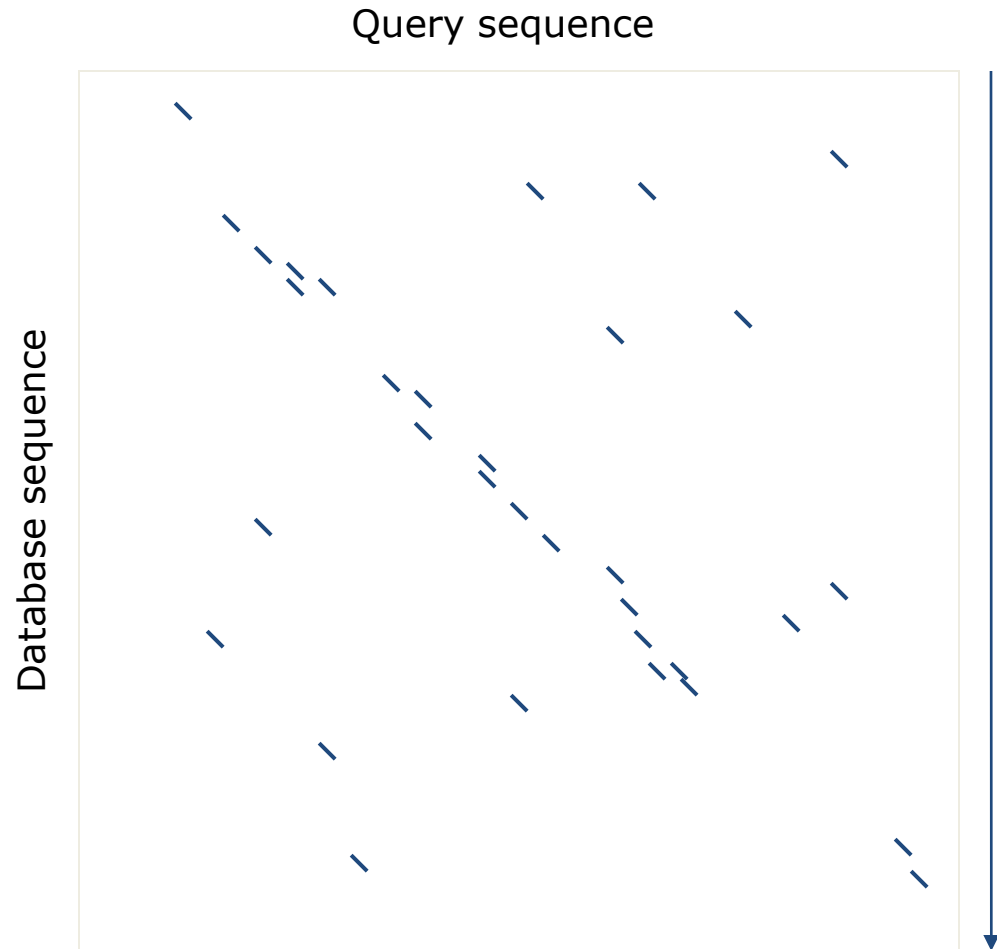
query
sequence

all overlapping
words of size 3

Hashing methods

**Scan the database,
looking up words in
the dictionary**

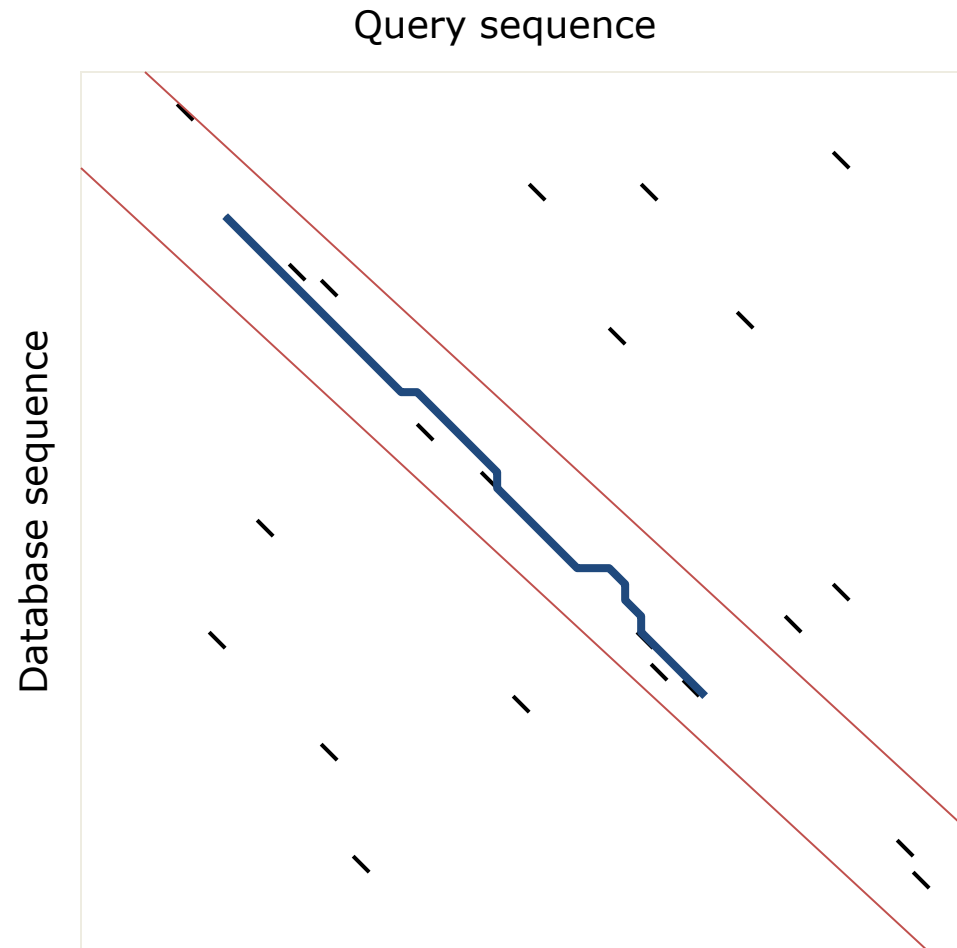
Use word hits to
determine where to search
for alignments
fills the dynamic
programming matrix
in $(N-2)+(M-2)$ operations
instead
of $M \times N$.



Hashing methods

**Scan the database,
looking up words in
the dictionary**

Use word hits to
determine where to search
for alignments

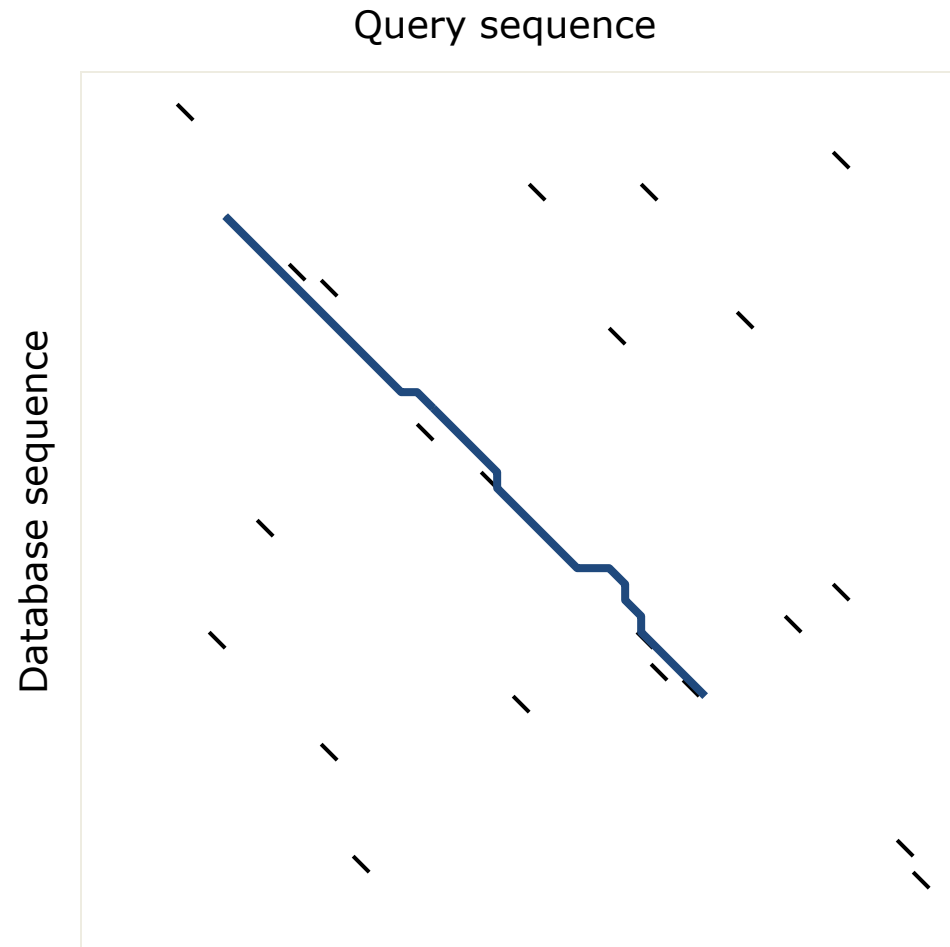


FASTA searches in a band

Hashing methods

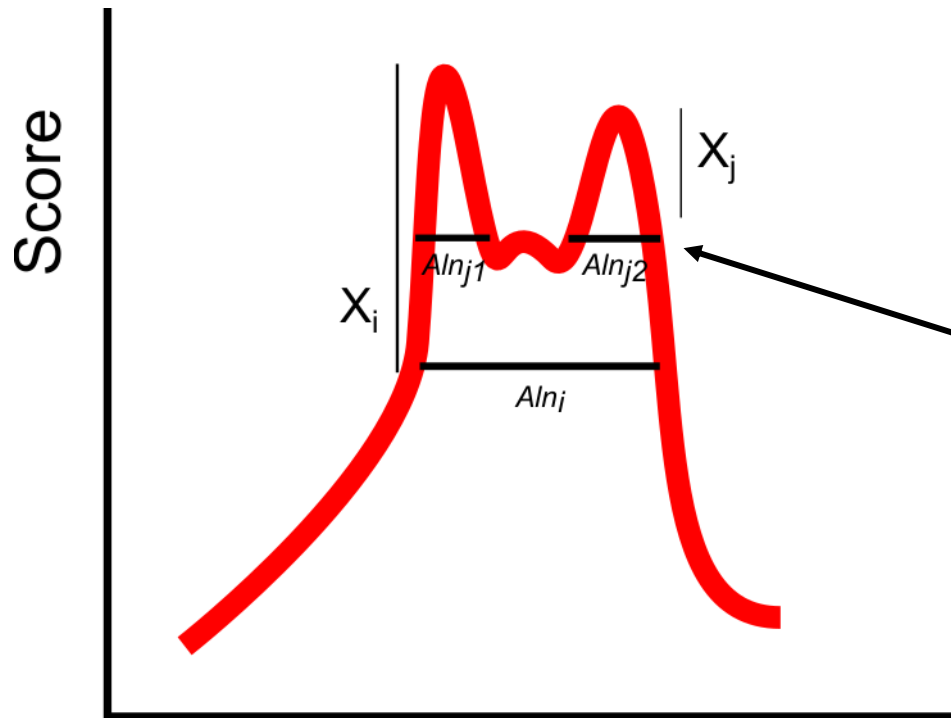
**Scan the database,
looking up words in
the dictionary**

Use word hits to
determine where to search
for alignments



BLAST: varios HSPs

En la jerga BLAST **un HSP es un “*high scoring pair*”** (o sea un alineamiento corto, local, de alto score, entre la secuencia “*query*” y alguna secuencia “*subject*” de la base de datos.)



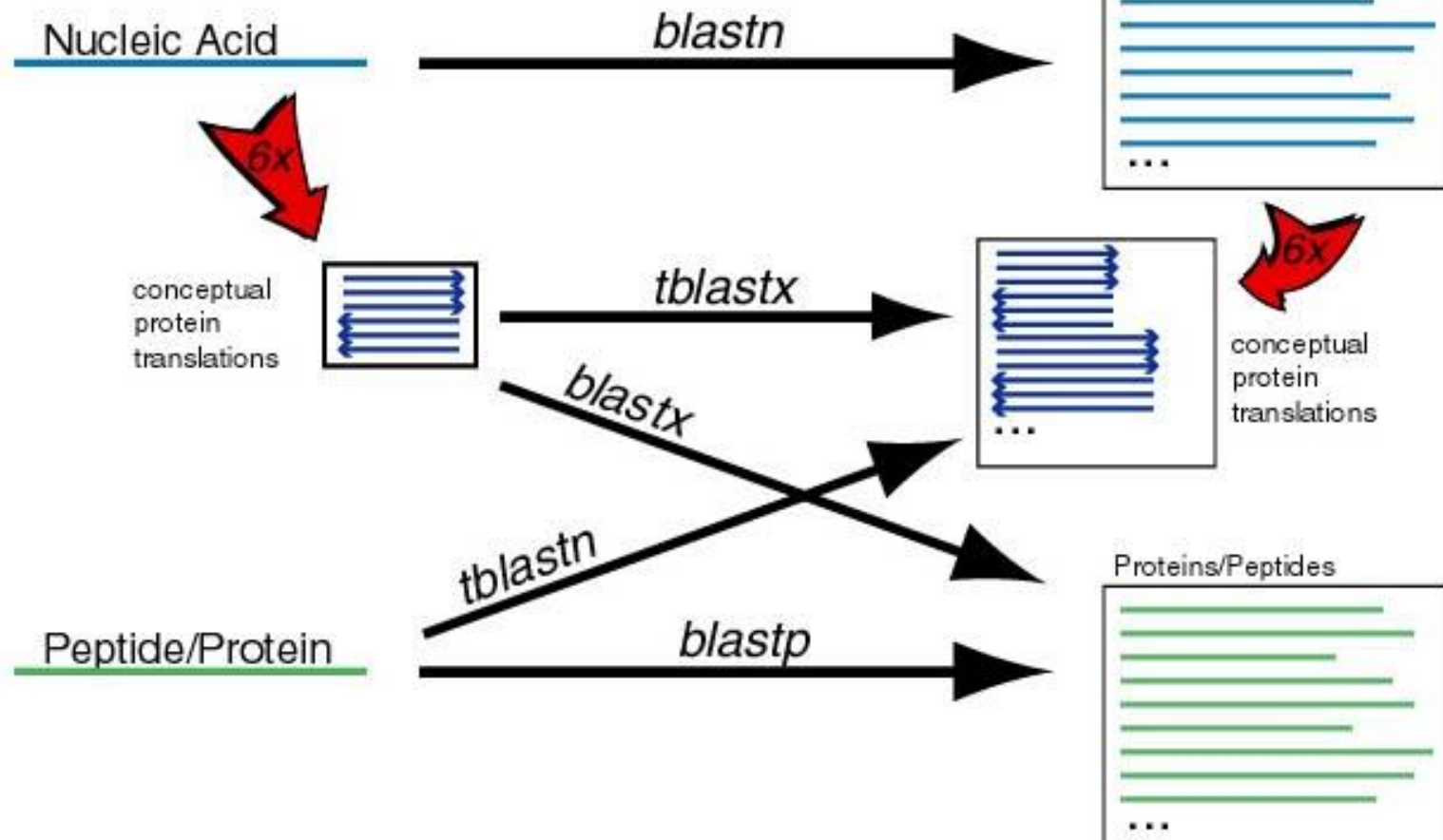
BLAST intenta extender el HSP, siempre que la caída del score sea menos que X (bits). Si lo logra, se repite con el próximo pico.

Si se aceptan caídas mayores del score (X_i), se logran alineamientos más largos (mas continuidad). Si solamente se aceptan caídas moderadas del score (X_j) BLAST entonces trunca los alineamientos (produce dos alineamientos o HSPs más cortos en este caso).

BLAST: algoritmos

QUERY
SEQUENCE

DATABASE



FASTA: algoritmos

- **FASTA**
 - protein-protein, DNA-DNA
- **fastx, fasty**
 - translated query, protein database
 - Permite frameshifts sólo entre codones (fastx) o dentro de un codón (fasty)
- **Ssearch**
 - Una implementación rigurosa del algoritmo de Smith-Waterman (sin heurísticas)
- **Prss**
 - Evalúa el significado de un alineamiento por permutación de una secuencia
- **Tfastx, tfasty**
 - Protein sequence vs DNA database

Evaluando alineamientos

- **Qué hacemos cuando estamos comparando dos secuencias que no son claramente similares, pero que muestran un alineamiento prometedor?**
- **Necesitamos un test de significancia**
- **Tenemos que responder a la pregunta:**
 - **Cuál es la probabilidad de que un alineamiento similar (con un score similar) ocurra entre proteínas no relacionadas?**

- **Generar secuencias al azar de la misma longitud y composición que la secuencia query y alinearlas**
 - Karlin & Altschul (1990); Altschul et al (1994); Altschul & Gish (1996)
- **Analizar la distribución de scores que se obtiene**

The Gumbel/Extreme value distribution

- In a database search (BLAST/FASTA) the alignment scores **do not** follow a normal/Gaussian distribution!

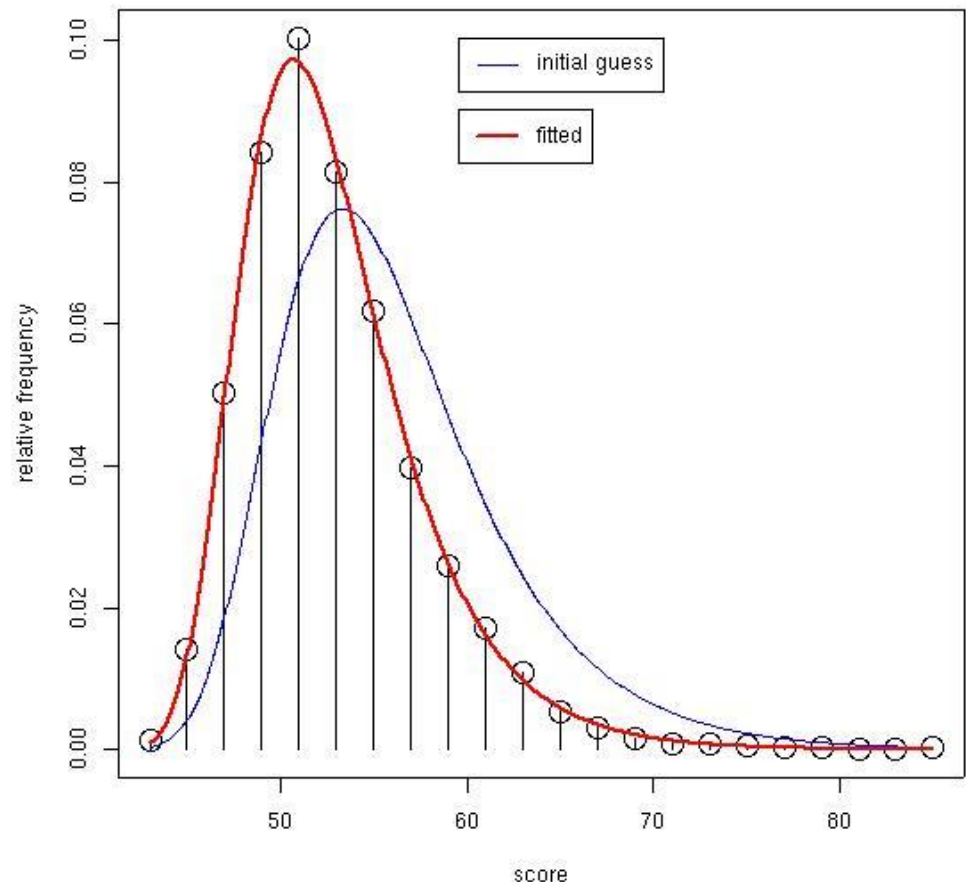
$$E = K m n e^{-\lambda S}$$

E is the number of alignments with a score = S

m,n: length of the sequences

K,λ: estimated parameters estimated (depend on the scoring matrix and the size of the database)

Extreme Value Distribution, Empirical method



E-value

Los hits pueden ser ordenados de acuerdo a su E-value o a su Score.

El E-value – más conocido como **EXPECT** value – es una función del score, el tamaño de la base de datos y de la longitud de la secuencia 'query'.

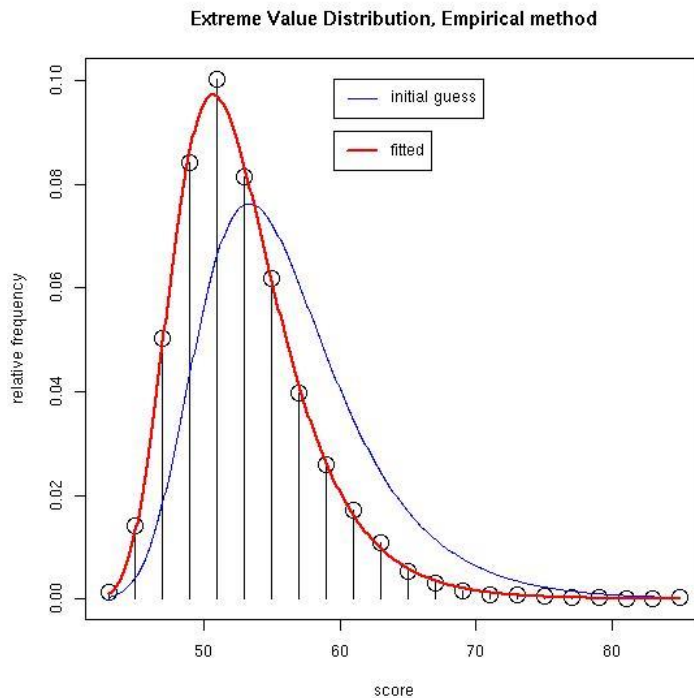
E-value: Número de alineamientos con un score = S que se espera encontrar si la base de datos es una colección de letras al azar.

Ejemplo: En el caso de un score=1 (un match o identidad) debería haber un número enorme de alineamientos. Uno espera encontrar menos alineamientos con un score de 5, 10, etc. Eventualmente, cuando el score es lo suficientemente alto, uno espera encontrar un número insignificante de alineamientos que sean debidos al azar.

Valores de E-value menores que $1e-6$ (1×10^{-6}) son generalmente muy buenos para proteínas, mientras que $E < 1e-2$ puede considerarse significativo. Es posible que un hit cuyo $E > 1$ sea biológicamente importante, aunque es necesario analizarlo más detalladamente para confirmarlo.

Observed vs expected

- Si la base de datos es suficientemente grande y contiene mayoritariamente secuencias no relacionadas la distribución de scores **observados** debería coincidir bastante con la distribución de scores **esperados** por azar (Pearson 1998)



Tamaño de la base de datos

$$E(S > x) = p(S > x) \text{ D}$$

- El número de alineamientos con un score $> S$ se incrementa linealmente con el tamaño de la base de datos
- \Rightarrow una secuencia (un alineamiento con un score S) encontrada en una búsqueda contra un genoma bacteriano con 1000-5000 secuencias va a ser 50-250 veces más significativa que un alineamiento con exactamente el mismo score en una base de datos como OWL (250,000 secuencias)
- Sin embargo, vimos que la base de datos tiene que ser **suficientemente grande** como para poder estimar P y E
- \Rightarrow Compromiso

Tamaño de la base de datos: un ejemplo

- **Objetivo:** encontrar el homólogo en *E. coli* de la DAHP synthase de *B. subtilis*
- ***E. coli* proteome**
 - **kdsA, $E(4283) < 0.00015$**
- **Swissprot**
 - **kdsA, $E(74417) < 0.0017$**
- **OWL**
 - **kdsA, $E(260784) < 0.0085$**
- **El mismo alineamiento, con el mismo score es 50 veces más significativo en la base de datos más chica.**

Identificar homólogos con eficiencia

- **Buscar en bases de datos pequeñas primero**
- **Repetir la búsqueda en una base de datos pequeña con un algoritmo más sensible (fasta3 con ktup 1 o ssearch)**
- **Si no hay hits significativos, buscar bases de datos más grandes, como nr (GenPept, TrEMBL)**

Límites de la estadística

- **En ciertos casos, la estadística de los alineamientos falla**
 - Lo que falla son las suposiciones que hicimos para llegar al modelo estadístico que describe - en este caso - la distribución de scores entre secuencias no relacionadas
- **En general se obtienen estimaciones incorrectas de E cuando**
 - Se usan penalidades de gap incorrectas
 - Existen regiones de baja complejidad en la secuencia query

Evaluando la estadística

```
opt      E()
< 20    13    0:=
22      0     0:
24      0     0:
26      0     0:
28      1     3:*
30     11    19:*
32     46    75:***
34    242   204:*****
36    493   419:*****
38    788   692:*****
40   1055   965:*****
42   1275  1180:*****
44   1299  1302:*****
46   1251  1326:*****
48   1186  1269:*****
50   1077  1158:*****
52    907  1018:*****
54    849   870:*****
56    714   727:*****
58    570   596:*****
60    456   483:*****
62    393   387:*****
64    313   308:*****
66    268   243:*****
68    219   192:*****
70    191   150:*****
72    127   117:*****
74     93    91:*****
76     91    71:*****
78     44    55:***
80     33    43:=*
82     22    33:=*
84     32    26:=*
86     19    20:=*
88     19    16:=*
90      8    12:=*
92      8     9:=*
94      5     7:=*
96      2     6:=*
98      3     4:=*
100     1     3:=*
102     3     3:=*
104     0     2:=*
106     1     2:=*
108     0     1:=*
110     0     1:=*
112     0     1:=*
114     0     1:=*
116     0     0:=*
118     1     0:=*
>120    7     0:=

one = represents 22 library sequences

inset = represents 1 library sequences
```

Mirar el histograma de
scores esperados y
observados

Mirar el E de la
secuencia no
relacionada con mayor
score

Evaluando la estadística (cont)

```

opt      E()
< 20    13    0:-
22      0    0:-
24      1    0:-
26      0    0:-
28      1    3:*
30     10   20:*
32     21   76:- *
34    105  205:---- *
36    272  422:----- *
38    540  697:----- *
40    937  972:----- *
42   1269 1188:----- *
44   1645 1311:----- *
46   1666 1335:----- *
48   1577 1278:----- *
50   1310 1166:----- *
52   1056 1025:----- *
54    851  876:----- *
56    669  732:----- *
58    423  601:----- *
60    419  487:----- *
62    255  390:----- *
64    196  310:----- *
66    181  245:----- *
68    154  193:----- *
70     99  151:----- *
72     74  118:----- *
74     63   92:----- *
76     60   72:----- *
78     47   56:----- *
80     48   43:----- *
82     36   33:----- *
84     33   26:----- *
86     27   20:----- *
88     21   16:----- *
90     18   12:----- *
92     20    9:----- *
94     20    7:----- *
96     17    6:----- *
98      7    4:----- *
100    10    3:----- *
102    11    3:----- *
104    10    2:----- *
106    11    2:----- *
108     7    1:----- *
110    10    1:----- *
112     6    1:----- *
114     4    1:----- *
116    11    0:----- *
118    10    0:----- *
>120   70    0:----- *
-----

```

one = represents 28 library sequences

inset = represents 2 library sequences

Si los histogramas Obs vs Exp coinciden

Y si el E del mejor alineamiento no relacionado es ~1

La estimaciones estadísticas están funcionando bien

Buscando homólogos en los límites

- **Secuencias homólogas distantes a menudo no tienen similitud estadísticamente significativa**
- **Secuencias con regiones de baja complejidad pueden tener similitud estadísticamente significativas, aunque no sean homólogas**
- **Secuencias homólogas generalmente son similares sobre toda la longitud de la secuencia o de un dominio**
- **Secuencias homólogas comparten un ancestro común**
 - **Si hay homología entre A y B; entre B y C; y entre C y D, A y D deben ser homólogos, aun cuando no muestren similitud estadísticamente significativa**

Low complexity sequences

- **Secuencias (o sub-secuencias) con bajo contenido de información**
 - AAAAAAAAAAAAAAAAAAAAAA
 - CAACAACAACAACAACAA
- **Secuencias con sesgo en la composición de bases (nucleótidos) o residuos (aminoácidos)**
- **Por el bajo contenido de información y el sesgo en la composición, suelen dar falsos positivos en las búsquedas por similitud**
 - PolyQ: proteínas con trectos de polyglutamina no están relacionadas por ancestría
 - Ej OTX2 (Transcription factor); CREB-binding protein (connects proteins with different functions); MED15/GAL11 (Subunit of the RNA polymerase II mediator complex)

Low complexity sequences

ORIGIN

```
1 maenlldgpp npraklssp gfsandstfd gslfdlendl pdelipngge lgllnsgnlv
61 pdaaskhkhql sellrggsgs sinpgignvs asspvqqglg gqaaggpnas nmaslsamgk
121 splsqgdssa pslpkqaast sgtpaasqa lnpqaqkvg latsspatsq tpggicmnan
181 fnqthpglln snsghsling asgggaqvmn aslgaarqr gaampvtpa mggassyla
```

ORIGIN

```
1 mmsylkqppy avnglsltts gmdllhpsvg ypatprkqrr erttfttraql dvlealfakt
61 rypdifmree valkinlpes rvqvwfknrr akcrqqqqqq qnggqnkvrp akkksspare
121 vssesgtsgq ftpssstsvp tiasssapvs iwspasispl sdplstsssc mqrssypmtyt
181 qasgysqgya gtsyfggmd cgsyltpmhh qlpgpgatls pmgtnavtsh lnqspaslst
241 qygasslgf nsttdcldyk dqtaswklfn nadcldykdg tsswkfqvl
```

//

```
721 mnsfnpmisg nvqlpqapmg praaspmnhs vqmnsmsgsvp gmaispsrmp qppnmgaht
781 nmmaaqapaq sqflpqnfq ssgamsvgm gppaqtgvs qgqvpgaalp nplnmlgpqa
841 sqlpcppvtq splhptpppa staagmpslq httpgmtpp qpaaptqpst pvsssgqtpt
901 ptpgsvpsat qtqstptvqa aaqaqvtpq qtpvqppsva tpqssqqqt pvhaqqpgtp
961 lsqaasidn rvptpsvas aetnsqqpgp dvpvlemkte tqaedtepd geskgeprse
1021 mmeedlqgas qvkeetdiae qksepmevde kkpevkvevk eeeessngt asqstpsqp
1081 rkkikfpeel rgalmptlea lyrqdpeslp frqpvdqll gipdyfdvnt npmdlstikr
1141 kldtgqyqep wqyvddvwlw fnnawlynrk tsrvykfcsk laevfeqid pvmqslgycc
1201 grkyefspqt lccygkqlct iprdaayysy qnryhfcekc fteiqgenvt lgddpsqpqt
1261 tiskdqfekk kndtldpepf vdckecgrkm hqicvlhydi iwpsgfvcdn clkktgrprk
1321 enkfsakrlq ttrlgnhled rvnkflrrqn hpeagevfvv vvasdktve vkgpmksrfv
1381 dsqemsesfp yrtkalfafe eidgvdvcff gmhvqeygsd cpppntrrvy isylsiahff
1441 rprclrtavy heiligyley vkklgyvtgh iwacppsgd dyifhchppd qkipkprlq
1501 ewykmldka faerihdyk difkqatedr ltsakelpyf egdfwpnvle esikeleqee
1561 eerkkeesta asettegsqg dsknakkkn kktknkssi srankkkpsm pnvsnslsqk
1621 lyatmekhke vffvihlhag pvintlppiv dpdpllsdcl mdgrdafllt ardkhwefss
1681 lrrskwstlc mlvelhtqgq drfvytcnec khhvetrwhc tvcedydlci ncyntkshah
1741 kmvkwglgld degssqgepq skspqesrrl siqrqiqlsv hacqcrnanc slpscqkmkr
1801 vvqhtkgckr ktnggcgvck qlialccyha khcqnckpv pfclnikhkl rqqqihrlq
1861 qaqlmrrrma tmntrnvppq slpsptsapp gtptqqpst qtpqppaqp pspvsmaspag
1921 fpsvartqpp ttvstgkpts qvpappppaq pppaaveaar qiereaaqqq hlyrvninns
1981 mppgrtgmgp pgsqmapvsl nvprpnqvsg pvmpsmppgq wqqaplpqq pmpglprpvi
2041 smqaqaavag prmpsvqppr sispsalqdl lrtlkspssp qqqqqvlnil ksnplmaaf
2101 ikqrtakyva nqpgmqppq lqsqpgmqpp pgmhqqpslq nlnamqagvp rpgvppqqqa
2161 mgglnpqgga lnimnpghnp nmasmnpqyr emlrqlqlq qqqqqqqqqq qqqqqqgsag
2221 maggmaghgq fqqpqqpggy ppamqqqqgm qqlhplqgss mgqmaaqmgq lgqmqgpglg
2281 adstoniqa laqrilaaaa mkaqiaspaa opomsaaahm lsaaqaashl paaqiatls
```

```
lmd intlnggssd tadkirihak nfeaalfaks
vta aaannnikpv eqhhinnlkn sgnannmnv
qqq qqqqqqqqrr qltpqqqqlv nqmkvapipk
ltp qdmeaakevy kihqqllfka rlqqqqaq
mqp pnssannnpl qqqssqntvp nvlqninqif
mte pvkqsfirky inqalrkiq alrdvknenn
nnn dtiatsatpn aaafsqqqna ssklyqmqqq
qaq aqaqaqaq aqaqaqaq aqaqaqaq
akd vevikqlsld asktnlrld vtlnlsneek
tkn enflkevql rifvkeilek caegifvkl
lrq qqmannnngn pgttstgnnn niatqqnmqq
qqq qqqqqqhiyp sstpgvanys amanapgnni
aat pslnktingk vngtrksnti pvtsipstnk
nps plktqtktngt pnpnmktvq spmgaqpsyn
rfk hrqEIFKdsp mdlfmstlgd clgikdeeml
ard qdsidisikd nklvmkskfn ksnrsysial
tss nmdvgnprkr kasvleispq dsiasvlspd
sek qevtneapfl tsgtsseqfn vwdwnnwtsa
```

BLAST compositional adjustment of scoring matrices

Las matrices de scoring (ej BLOSUM62) son derivadas a partir de alineamientos de proteínas globulares, con una

composición de aminoácidos definida

Pero muchas veces las secuencias “query” tienen composiciones de aminoácidos con sesgos muy marcados y diferentes a las de las secuencias utilizadas para derivar las matrices

Ej proteínas hidrofóbicas, Cys-rich, proteínas codificadas por genomas con alto sesgo (AT rich, GC rich), que afectan los codones más utilizados

En esos casos hay maneras de ajustar las matrices (ej recalcular BLOSUM62 on-the-fly) para que funcionen mejor frente a estos casos


BLAST compositional adjustment of scoring matrices

Query = human insulin NP_000198
Program = blastp
Database = *C. elegans* RefSeq

Option = NO compositional adjustment

Scoring Parameters

Matrix	BLOSUM62
Gap Costs	Existence: 11 Extension: 1
Compositional adjustments	Conditional compositional score matrix adjustment



```
>[ref|NP_501926.1] UG INSulin related family member (ins-1) [Caenorhabditis elegans]
Length=109

Score = 34.7 bits (78), Expect = 0.009
Identities = 30/100 (30%), Positives = 41/100 (41%), Gaps = 14/100 (14%)

Query 11  LALLALWGPDPAAAFVNQHLCGSHLVEALYLVCGERGFFYTPKTRREAEDLQVGQVELGG 70
          LA+L L P P+ A + LCGS L L VC + +R A+
Sbjct 17  LAILLSSPTPSDASIR--LCGSRLTTTLLAVCRNQLCTGLTAFKRSADQSY----- 66

Query 71  GPGAGSLQPLALEGSLQKRG-IVEQCCTSI CSLYQLENYC 109
          A + + L QKRG I +CC CS L+ +C
Sbjct 67  ---APTTRDLFHIHQKRGGIATECCEKCSFAYLKTF 103
```

Option = conditional compositional score matrix adjustment

```
>[ref|NP_501926.1] UG INSulin related family member (ins-1) [Caenorhabditis elegans]
Length=109

Score = 33.5 bits (75), Expect = 0.020, Method: Compositional matrix adjust.
Identities = 27/100 (27%), Positives = 39/100 (39%), Gaps = 12/100 (12%)

Query 10  LLALLALWGPDPAAAFVNQHLCGSHLVEALYLVCGERGFFYTPKTRREAEDLQVGQVELG 69
          LA+L L P P+ A + LCGS L L VC + +R A+
Sbjct 16  FLAILLSSPTPSDASIR--LCGSRLTTTLLAVCRNQLCTGLTAFKRSADQ-----S 65

Query 70  GPGAGSLQPLALEGSLQKRGIVEQCCTSI CSLYQLENYC 109
          P L + ++ GI +CC CS L+ +C
Sbjct 66  YAPTTRDL--FHIHQKRGGIATECCEKCSFAYLKTF 103
```

BLAST compositional adjustment of scoring matrices

Fig. 1 Examples of alignment extensions yielded by compositional adjustment of the scoring system. The sequences ...

(a)

```
78 ISGAILFEETLFQKNEAGVPMVNLLHNENIIPGIKVDKGLVNIPCTDEE--KSTQGLDGLAERCKEYYKAGA 147
I GAILFE+T+ K ++ L + ++P +K+DKGL ++ + K L L +R E + G
67 ILGAILFEQTMSKIDGKYTADFLWEEKKVLFPFLKIDKGLNDLDADGVQTMKPNPTLADLLKRANERHIFG- 137

148 RFAKWRTVLVIDTAKGKPTDLS-IHETAWGLARYASICQQNRLVPIVEPEI 197
K R+V+ K+ P ++ + E + +A A + L+PI+EPE+
138 --TKMRSVI----KKASPAGIARVVEQQFEVA--AQVVAAG-LIPIIEPEV 179
```

(b)

```
78 ISGAILFEETLFQKNEAGVPMVNLLHNENIIPGIKVDKGLVNIPCTDEE--KSTQGLDGLAERCKEYYKAGA 147
I GAILFE+T+ K + L + + ++P +K+DKGL ++ + + K L L +R+ E + G
67 ILGAILFEQTMSKIDGKYTADFLWEEKKVLFPFLKIDKGLNDLDADGVQTMKPNPTLADLLKRANERHIFG- 137

148 RFAKWRTVLVIDTAKGKPTDLS-IHETAWGLARYASICQQNRLVPIVEPEILADGPHSIEVCAVVTQKVLSC 218
K+R+V+ K+ P ++ + E + +A A ++ L+PI+EPE+ ++ ++ C + + +
138 --TKMRSVI----KKASPAGIARVVEQQFEVA--AQVVAAG-LIPIIEPEVDINNVDKVQ-CEEILRDEIRK 199

219 VFKALQE-NGVLLEGALLKPNMVTAGYECTAKTTTQDVGFLTVRTLRRTVPPALPGVVFLSGGQSEEEAS 287
+ AL E ++V+L+ L P + E T P + VV LSGG S E+A+
200 HLNALPETS NVMLKLTL--PTVENLYEEFTKH-----PRVVRVVALSGGYSREKAN 248
```

(c)

```
78 ISGAILFEETLFQKNEAGVPMVNLLHNENIIPGIKVDKGLVNIPCTDEE--KSTQGLDGLAERCKEYYKAGA 147
I GAILFE+T+ K + L + + ++P +K+DKGL ++ + + K + L L +R+ E + G
67 ILGAILFEQTMSKIDGKYTADFLWEEKKVLFPFLKIDKGLNDLDADGVQTMKPNPTLADLLKRANERHIFG- 137

148 RFAKWRTVLVIDTAKGKPTDLS-IHETAWGLARYASICQQNRLVPIVEPEILADGPHSIEVCAVVTQKVLSC 218
K+R+V+ K+ P ++ + E + +A A ++ L+PI+EPE+ ++ ++ ++ +++
138 --TKMRSVI----KKASPAGIARVVEQQFEVA--AQVVAAG-LIPIIEPEVDINNVDKVQCEEILRDEIRKH 200

219 VFKALQENGVLLEGALLKPNMVTAGYECTAKTTTQDVGFLTVRTLRRTVPPALPGVVFLSGGQSEEEAS 287
+ + ++V+L+ L P + + E T P + VV LSGG S E+A+
201 LNALPETS NVMLKLTL--PTVENLYEEFTKH-----PRVVRVVALSGGYSREKAN 248
```

a) standard BLOSUM-62 substitution matrix

b) composition-adjusted matrix derived from BLOSUM-62 with unconstrained relative entropy

b) composition-adjusted matrix derived from BLOSUM-62 with relative entropy constrained to 0.566 bits

Bioinformatics. Sequence and Genome analysis. David W Mount, CSHL Press (2001)

Hugues Sicotte (NCBI). (slides DP)

Javascript-based implementations for various algorithms
<http://rna.informatik.uni-freiburg.de/Teaching/>