

Alineamiento de pares de secuencias de DNA y proteína - introducción

- Dadas 2 o más secuencias, lo que generalmente deseamos es:
 - cuantificar su grado de similitud
 - determinar las correspondencias evolutivas (homología) residuo - residuo
 - describir e interpretar patrones de conservación y variación
 - inferir las relaciones evolutivas entre las secuencias
- Para definir índices cuantitativos de similitud entre secuencias necesitamos primero definir las **correspondencias evolutivas (homología)** entre los **residuos** de distintas secuencias, en forma de un **alineamiento**. Este representa una de las herramientas básicas de la bioinformática y biología evolutiva
- Para optimizar un alineamiento necesitamos acomodar las correspondencias entre residuos idénticos, distintos, inserciones y deletiones. Esto se logra matemáticamente usando **factores de ponderación** ("weightings") para cada caso. Así un match tiene un peso, un mismatch otro y los indeles un tercer valor. Dos secuencias se comparan residuo a residuo, generándose un valor de puntuación (**score**) acorde a estas ponderaciones, que refleja el nivel de similitud entre ellas

Tema II: alineamientos pareados y múltiples - determinación de correspondencia de homología en secuencias moleculares

1.- Alineamientos pareados

- evolución de secuencias y clasificación de mutaciones
- indeles y gaps
- alineamientos globales (Needleman-Wunsch) vs. locales (Smith-Waterman);
- programación dinámica;
- dot plots;
- matrices de costo de sustitución, penalización de gaps y cuantificación de la similitud;
- evaluación estadística de la similitud entre pares de secuencias;
- escrutinio de bases de datos mediante BLAST: Búsquedas a nivel de DNA vs. AA;
- la familia BLAST e interpretación de resultados de búsquedas de secuencias homólogas
- prácticas usando NCBI BLAST y DOE-IMG BLAST

```

>F01715488561ref12P_08669120.1 Translation elongation factor G:small GTP-binding protein domain
[Nitrosomonas eutropha C71]
g1:171494077gb:K018626.1 Translation elongation factor G:small GTP-binding protein domain
[Nitrosomonas eutropha C71]
length=696

Score = 828 bits (2140), Expect = 0.0
Identities = 434/697 (62%), Positives = 541/697 (77%), Gaps = 9/697 (1%)

Query 1  MTRFSLRTNIGIMHIDAGKTTTTRVLTTRIKITGTHGASQMMHQAQKQERG 60
          M++ LE+ RNIGIMHIDAGKTTT+R+L+TTO EK+GE H+GA+ MNM QKQERG
Sbjct 1  MTRFSLRTNIGIMHIDAGKTTTTRVLTTRIKITGTHGASQMMHQAQKQERG 60

Query 61  XXXXXXXXXXXXXXXX-----DRIINIITDPORVDTFVERSLRVLDGAVLDGQVE 113
          ITTSAATT W +HRIIN+ITDPORVDTF+VERSLRVLDGA V + GV+
Sbjct 61  ITTSAATTCPKMMNTSEERINVIDTPORVDTFVERSLRVLDGACTVCVGQV 120 (... truncado)
  
```

Homología entre pares de secuencias de DNA: conceptos y terminología básica

- A lo largo de la evolución las secuencias descendientes de otra ancestral van acumulando diversos tipos de mutaciones. Estas son **mutaciones puntuales** o **reorganizaciones genómicas**, que pueden involucrar **inserciones, deletiones, inversiones, translocaciones o duplicaciones**, mediados por distintos mecanismos de recombinación (homóloga e ilegítima)
- Cualquier análisis filogenético y/o evolutivo de secuencias moleculares requiere de un **alineamiento** para poder comparar sitios homólogos entre las secuencias a estudiar. Para ello se escriben las secuencias en filas una sobre la otra, de modo que los sitios homólogos quedan alineados por columnas. Cada sitio o columna del alineamiento corresponde a un **carácter**, y los nt o aa que ocupan dichas posiciones representan los distintos **estados del carácter**

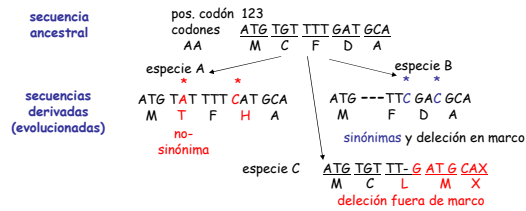
Tema II: alineamientos pareados y múltiples - determinación de correspondencia de homología en secuencias moleculares

- Alineamientos pareados vs. alineamientos múltiples (AMs)
- complejidad algorítmica
- estrategias para la generación de AMs
- AMs de DNA vs AA y su relación con la filogenética
- inferencias estructurales de AMs
- aplicaciones de AMs para la búsqueda de homólogos en bases de datos [PSI-BLAST; perfiles y cadenas ocultas de Markov (HMMs)]
- prácticas con:
 - la familia clustalW, clustalX, T-coffee y muscle ; uso del servidor RevTrans
 - formatos de secuencias y su interconversión (ReadSeq)

Homología entre pares de secuencias de DNA: conceptos y terminología básica

- Cuando por eventos de inserción o deletión (**indeles**) las secuencias homólogas presentan distintas longitudes, es necesario introducir "gaps" en el alineamiento para mantener la correspondencia entre sitios homólogos situados antes y después de las regiones afectadas por indeles. Estas regiones se identifican mediante guiones (-). **Los indeles no se distribuyen aleatoriamente en las secuencias codificadoras**. Casi siempre aparecen ubicados entre dominios funcionales o estructurales, preferentemente en bucles (loops) que conectan a dichos dominios. Esto vale tanto para RNAs estructurales (tRNAs y rRNAs) como para proteínas. No suelen interrumpir el marco de lectura.

Homología entre pares de secuencias de DNA: tipos de mutaciones en secs. codificadoras de proteínas



- Todas las mutaciones en 2^{da} posiciones resultan en sustituciones no sinónimas
- 96% de mutaciones en 1^{ra} posiciones resultan en sustituciones no sinónimas
- Casi todas las sustituciones sinónimas ocurren en las 3^{as} posiciones
- las deleciones o inserciones en secs. codificadoras de aa suceden generalmente en múltiplos de tres nt; de no ser así se generan cambios de marco de lectura corriente abajo de la mutación, con frecuencia generando un pseudogen no funcional

Programación dinámica y la generación de alineamientos pareados (globales y locales)

- Pares de secuencias pueden ser comparados usando **alineamientos globales y locales**, dependiendo del objetivo de la comparación.

Un **alineamiento global** fuerza el alineamiento de ambas secuencias a lo largo de toda su longitud. Usamos aln. globales cuando estamos seguros de que la homología se extiende A lo largo de todas las secuencias a comparar. Este es el tipo de alineamientos que generan programas de alineamiento múltiple tales como clustal, T-Coffee o muscle.

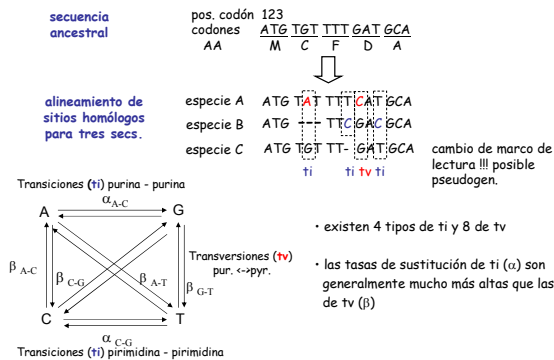
(a)

P00001	1	MGUVERGKGIIFMKSCQSTVEKGGKXKXTPHLSGLFGRKTOQAPGYSSTAAENK----	58
		D EG+ +F QC T + K+ GP L G+ GRK G A G++Y+ N N G+	
P00090	1	Q-DAARGAVT-----KQNTCHRADKXNVGALGVGVGRKAGTAAGPTTFLNHSGEAGL	56
P00001	59	IWGEDTLMRYLNPVKYIP-----GTXMIFVGIKKKEERADLIAYLKATNE	105
		+W ++ ++ YL +P Y+ TEN F + ++R D+ AYI AT +	
P00090	57	VWQENIATLPDPNATLKKFLTDGQADKATGSKTNTT-KLANDQRRKVAATL--ATLK	114

Alineamiento global óptimo del citocromo C humano (105 residuos, SWISS-PROT acc. P00001) y citocromo C2 de Rhodospseudomonas palustris (114 residuos, SWISS-PROT acc. P00090).

La matriz de puntuación o ponderación ("scoring matrix") empleada fue **BLOSUM62**, con **costo de gaps** afines de $-(11 + k)$. La puntuación del alineamiento global es de 131, usando el algoritmo de **Needleman-Wunsch**.

Homología entre pares de secuencias de DNA: alineamiento y tipos de mutaciones



Transiciones (ti) purina - purina

Transversiones (tv) pur. ↔ pyr.

Transiciones (ti) pirimidina - pirimidina

• existen 4 tipos de ti y 8 de tv

• las tasas de sustitución de ti (α) son generalmente mucho más altas que las de tv (β)

Programación dinámica y la generación de alineamientos pareados (globales y locales)

Un **alineamiento local** sólo busca los segmentos con la puntuación más alta. Se usa por ejemplo en el escrutinio de bases de datos de secuencias debido a que la homología entre pares de secuencias puede a veces existir sólo a nivel de ciertos dominios, pero no a lo largo de toda la secuencia (**estructura modular de proteínas; genes discontinuos intrones-exones; barajado de exones...**).

BLAST y FASTA buscan alineamientos locales con alta puntuación (HSPs ó high-scoring pairs)

(b)

P13569	1221	EGGNALLENISFISPGQVLLGRTSGSKSTLLSAFLRLI-----NYERQIDQVSE	1273
		+ ++ +S ++ G+ + L+G +GSGKS +A L +L T GEI DG	
P33593	13	QAAQPLVHGVSLTLQRGVLLALVGGSGSGKSLTCAATLGLIPAGVQTAGEILADGPK	70
P13569	1274	WDSITL-----QWNRKAFGVIPKVFIPSTPTREKNDPTQNSQIINKVADEV	1322
		L Q R AF + + + + + E AD+	
P33593	71	VSPCALRGIKIATIMQNFSAFNPFI-----RTMTHARETCLALGKPADDA	116
P13569	1323	GLRSVIRQFPF-KGLDFVLVDGCGVLSHGKGLNCILARSVLSKATILLDEPSANLDPV	1379
		L + IE VL +S G Q M +A +VL ++ ++ DEP+ LD V	
P33593	117	TLTAATRAVLENAARVILKLYPFNSGHLQNMIAHAYLCSPTFIADPTTOLDV	174

Alineamiento local óptimo del regulador de conductancia transmembranal de fibrosis cística de humano (1480 residuos, SWISS-PROT acc. P13569) y la proteína transportadora de Ni dependiente de ATP de E. coli (253 residuos, SWISS-PROT acc. P33593).

La matriz de puntuación o ponderación ("scoring matrix") empleada fue **BLOSUM62**, con **costo de gaps** afines de $-(11 + k)$. La puntuación del alineamiento local es de 89, usando el algoritmo de **Smith-Waterman**.

Programación dinámica y la generación de alineamientos pareados (globales y locales)

- Estudiar el fundamento de los **algoritmos de PD** es un buen punto de arranque para entender lo que acontece dentro de software usado extensamente en biología computacional:

El corazón de programas como BLAST, FASTA, CLUSTALW, HMMER, GENSCAN, MFOLD y los de inferencia filogenética (PHYLIP, PAUP, MrBayes ...) emplean alguna forma de programación dinámica, con frecuencia **variantes heurísticas**

- **Alineamientos pareados: el problema visto desde la perspectiva biológica**

El supuesto básico es que si dos secuencias se parecen mucho a lo largo de sus secuencias es porque comparten un ancestro común: son homólogas. Es decir, **inferimos la homología a partir de la similitud**.

Para cuantificar objetivamente el nivel de similitud necesitamos un sistema de puntuación (scoring scheme) que lo refleje adecuadamente, desde una perspectiva evolutiva

El objetivo es **alinear las dos secuencias de tal manera que se maximice su similitud**

Para ello necesitamos un algoritmo, ya que no es práctico evaluar todos los alineamientos posibles entre un par de secuencias dado el elevadísimo número de combinaciones ($2^{2N}/(2TIN)^{1/2}$). Así para dos secs. de 300 residuos existen 10^{179} alns. posibles!!!

Los algoritmos de programación dinámica son adecuados para este trabajo

Programación dinámica y la generación de alineamientos pareados (globales y locales): dot plots y visualización de la similitud entre secuencias

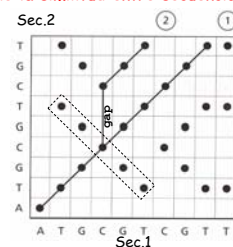
- las 2 secs. representan los dos ejes de la gráfica

- se pone un punto donde ambas coinciden

- la diagonal más larga representa la región de mayor identidad

- el camino 1 es el preferido al ser el más parsimonioso (implica menos cambios)

- la diagonal cruzada revela un **palíndromo**



alineamiento diagonal 1

secuencia 1: ATGCGTCTGTT
secuencia 2: ATGCGTCTGTT

alineamiento diagonal 2

secuencia 1: ATG---CGTCGTT
secuencia 2: ATGCGTCGT

Tema 2: Alineamientos pareados y búsqueda de homólogos en bases de datos mediante BLAST

Programación dinámica y la generación de alineamientos pareados (globales y locales): dot plots y visualización de la similitud entre secuencias

secuencia 1: ATGCGTCGTT
secuencia 3: ATCGTCAT

secuencia 1: ATGCGTCGTT
secuencia 3: ATCGTCAT

- la diagonal cruza celdas vacías, correspondientes a posiciones con distintos estados de carácter
- se pueden alinear dos secuencias aleatorias postulando una combinación de sustituciones y gaps
- se puede calcular el "costo" de un alineamiento contando el número de sustituciones (s) y gaps (g), o una función de ellos: p. ej.: $D = s + w$, donde w es un factor de penalización (FP) para la creación de gaps (**gap penalty**) donde para $w = 1$ abrir un gap cuesta igual que una sustitución $w = 2$ cuesta el doble un gap que una sustitución

Se emplean valores bajos de w si pensamos abundaron indeles en la hist. evol. de las secs.

- generalmente $w = g + hl$, donde l es la longitud del gap, g es un FP de apertura del gap, y h es el FP para extender el gap. Estos son **FP afines**. La fórmula es muy flexible al permitir un control independiente del número y longitud (l) de los gaps mediante g y h

Programación dinámica: algoritmo de Needleman-Wunsch y alineamientos pareados globales

Saul Needleman and Christian Wunsch (1970). *A general method applicable to the search for similarities in the amino acid sequence of two proteins*, J Mol Biol. 48(3):443-53.

Este algoritmo es un ejemplo de PD y **garantiza encontrar el alineamiento global de puntuación máxima**

La PD constituye una técnica muy general de programación. Se suele aplicar cuando existe un espacio de búsqueda muy grande y éste puede ser estructurado en una serie o sucesión de estados tales que:

- el estado inicial contiene soluciones triviales de subproblemas
- cada solución parcial de estados posteriores puede ser calculada por iteración sobre un número fijo de soluciones parciales de los estados anteriores
- el estado final contiene la solución final

Un algoritmo de PD consta de 3 fases:

- fase de inicialización y definición recurrente del score óptimo
- relleno de la matriz de PD para guardar los scores de subproblemas resueltos en cada iter. Se comienza por resolver el subproblema más pequeño
- un rastreo reverso de la matriz para recuperar la estructura de la solución óptima

alineamientos pareados y penalidad de gaps afines

Dado que un **sólo evento mutacional puede insertar o eliminar varios nucleótidos** de una secuencia, un indel largo no debe de ser penalizado mucho más que otro más corto ubicado en la misma región de un gen. De ahí el uso de **penalidades de gaps afines** (affine gap penalties or costs), que cobran una penalidad relativamente alta por abrir un gap y una penalidad más baja por cada posición sobre la que se extiende. La calidad de un alineamiento depende en gran medida de los **valores de apertura y extensión de gap** elegidos.

Programación dinámica y la generación de alineamientos pareados (globales y locales) - algoritmo de DP para alineamientos globales

Como ejemplo vamos a alinear dos palabras: COELACANTH y PELICAN usando el siguiente esquema de ponderación: match = 1; mismatch = -1; gap = -1

Existen dos alineamientos con el mismo score máximo:

COELACANTH
P-ELICAN--

COELACANTH
-PELICAN--

por tratarse de aln. globales, cada letra está alineada con otra o con un gap. Este no es el caso en aln. locales.

El alineamiento acontece en un arreglo bidimensional en la que cada celda corresponde al apareamiento de un residuo de cada secuencia

El alineamiento comienza arriba izda y sigue una trayectoria horizontal o vertical cuando hay un gap que introducir, y en la diagonal cuando tenemos apareamientos. Los gaps nunca se aparean entre ellos

Nótese que tenemos una fila y col. vacías adicionales

Programación dinámica: algoritmo de Needleman-Wunsch y alineamientos pareados globales

Un valor de puntuación es escogido para cada tipo de sustitución (par de residuos o aln. de residuo contra un gap). El set completo de estas puntuaciones conforman a una matriz de ponderaciones o puntuaciones (**scoring matrix**), de dimensiones $S(i,j)$

Existen muchas definiciones del score de un alineamiento, pero la más común es simplemente la suma de scores o puntuaciones para cada par de letras alineadas y pares letra-gap, que conforman el alineamiento.

Así, para la matriz de sustitución siguiente y un w lineal de 5, calcula la puntuación del siguiente alineamiento

	A	G	C	T
A	10	-1	-3	-4
G	-1	7	-5	-3
C	-3	-5	9	0
T	-4	-3	0	8

AGACTAGTTAC
CGA---GACGT

Score = $-3+7+10-3 \times 5 + 7-4+0-1+0 = 1$

Programación dinámica: algoritmo de Needleman-Wunsch y alineamientos pareados globales

En realidad no se guardan los caracteres en las celdas. Estas contienen dos valores: una puntuación (score) y un apuntador. El score se calcula a partir del esquema de puntuación o más generalmente, de una matriz de puntuaciones. El apuntador es un indicador de dirección (flecha) que apunta en una de tres direcciones: arriba, izquierda o en diagonal izda. hacia arriba.

I. Fase de inicialización

- se comienza asignando valores a la primera fila y columna. La siguiente fase del algoritmo depende de estas asignaciones.
- La puntuación de cada celda corresponde al "gap score" x distancia al origen
- Las flechas apuntan todas al origen, lo que asegura que los alineamientos puedan seguirse hasta el origen al final del algoritmo. Esto es un requisito para conseguir un aln. global

$F(i, 0) = i \times \text{gap penalty}$ $i = \text{pos columna}$
 $F(0, j) = j \times \text{gap penalty}$ $j = \text{pos fila}$

Programación dinámica: algoritmo de Needleman-Wunsch y alineamientos pareados globales

II. Fase de relleno o inducción.

- Se rellena toda la tabla con "scores" y apuntes, requiriéndose los valores de las celdas vecinas diagonal, vertical y horizontal. Por ello sólo se puede comenzar en la celda (1,1)
- Se calculan tres scores: uno de match, uno de gap horizontal y otro de gap vertical:

- El **match score** = score de la diagonal + puntuación de apareamiento (+1 ó -1)
- El **gap score horizontal** = score de celda izda + gap score
- El **gap score vertical** = score de celda superior + gap score
- Se asigna a la nueva celda el valor más alto de los tres y una flecha en dirección de la celda vecina con mayor score

$$F(i, j) = \begin{cases} F(i-1, j) + \text{gap-penalty} \\ F(i-1, j-1) + s(i, j) \\ F(i, j-1) + \text{gap-penalty} \end{cases}$$

	C	O	E	L	A	C	A	N	T	H
P	0	-1	-2	-3	-4	-5	-6	-7	-8	-10
E	-1	-1	-2	-3	-4	-5	-6	-7	-8	-10
L	-2	-2	-2	-3	-4	-5	-6	-7	-8	-10
I	-3	-3	-3	-3	-4	-5	-6	-7	-8	-10
C	-4	-4	-4	-4	-4	-5	-6	-7	-8	-10
A	-5	-5	-5	-5	-5	-5	-6	-7	-8	-10
N	-6	-6	-6	-6	-6	-6	-6	-7	-8	-10

- match score = 0 + (-1) = -1 → es el score más alto y por tanto va a la celda
- gap score horizontal = -1 + (-1) = -2
- gap score vertical = -1 + (-1) = -2
- la flecha apunta al 0 por ser el score vecino más alto

Programación dinámica: algoritmo de Needleman-Wunsch y alineamientos pareados globales

III. Fase de rastreo regresivo o hacia el origen

Para recuperar el alineamiento tenemos que regresarnos de la celda ubicada en el vértice de abajo a la dcha. y seguir el camino indicado por el puntero hasta el inicio

Dado que seguimos el camino del alineamiento óptimo del final hacia el principio, tenemos que revertir la secuencia al final del algoritmo para tenerla en la orientación correcta

	C	O	E	L	A	C	A	N	T	H
P	0	-1	-2	-3	-4	-5	-6	-7	-8	-10
E	-1	-1	-2	-3	-4	-5	-6	-7	-8	-10
L	-2	-2	-2	-3	-4	-5	-6	-7	-8	-10
I	-3	-3	-3	-3	-4	-5	-6	-7	-8	-10
C	-4	-4	-4	-4	-4	-5	-6	-7	-8	-10
A	-5	-5	-5	-5	-5	-5	-6	-7	-8	-10
N	-6	-6	-6	-6	-6	-6	-6	-7	-8	-10

Existen dos alineamientos globales con el mismo score máximo = 0

COELACANTH y COELACANTH
P-ELICAN-- y -PELICAN--

Programación dinámica: algoritmo de Needleman-Wunsch y alineamientos pareados globales

II. Fase de relleno o inducción.

- Segundo ciclo. Se continúa llenando la segunda fila o columna siguiendo las mismas reglas

$$F(i, j) = \begin{cases} F(i-1, j) + \text{gap-penalty} \\ F(i-1, j-1) + s(i, j) \\ F(i, j-1) + \text{gap-penalty} \end{cases}$$

	C	O	E	L	A	C	A	N	T	H
P	0	-1	-2	-3	-4	-5	-6	-7	-8	-10
E	-1	-1	-2	-3	-4	-5	-6	-7	-8	-10
L	-2	-2	-2	-3	-4	-5	-6	-7	-8	-10
I	-3	-3	-3	-3	-4	-5	-6	-7	-8	-10
C	-4	-4	-4	-4	-4	-5	-6	-7	-8	-10
A	-5	-5	-5	-5	-5	-5	-6	-7	-8	-10
N	-6	-6	-6	-6	-6	-6	-6	-7	-8	-10

El mejor score del alineamiento hecho hasta ahora tiene vale -2 y corresponde a:

CO ó CO
-P ó P-

- match score = -1 + (-1) = -2 → es el score más alto y por tanto va a la celda
- gap score horizontal = -1 + (-1) = -2
- gap score vertical = -2 + (-1) = -3
- la flecha puede apunta al -1 diagonal u horizontal. Se toma una decisión arbitraria pero consistente si se vuelve a dar el caso (p. ej. aceptar siempre diagonal)

Programación dinámica: algoritmo de Smith-Waterman y alineamientos pareados locales

Smith TF, Waterman MS (1981) J. Mol. Biol 147(1):195-7

- Se trata de una modificación simple del algoritmo de Needleman-Wunsch. Sólo hay tres cambios:

- La 1a. fila y columna es inicializada con ceros, en vez de gap penalties incrementales
- El score máximo no es nunca < 0 y sólo se guardan apuntes en las celdas si su score es > 0
- El rastreo reverso comienza desde la celda con el score más alto de la tabla (y no de la última celda de la misma) y termina en una celda con score 0 (y no en la primera)

- Estas modificaciones tienen un profundo efecto sobre el comportamiento del algoritmo, y como resultado obtenemos el alineamiento local con mayor puntuación de todos los posibles en la matriz.

Programación dinámica: algoritmo de Needleman-Wunsch y alineamientos pareados globales

II. Fase de relleno o inducción.

- Segundo ciclo. Se continúa llenando la segunda fila (o columna) siguiendo las mismas reglas y una vez llena, se continúa con la tercera fila (o columna) hasta terminar de llenar la tabla siguiendo la expresión:

$$F(i, j) = \max \{ F(i-1, j-1) + s(i, j), F(i-1, j) + \text{gap-penalty}, F(i, j-1) + \text{gap-penalty} \}$$

	C	O	E	L	A	C	A	N	T	H
P	0	-1	-2	-3	-4	-5	-6	-7	-8	-10
E	-1	-1	-2	-3	-4	-5	-6	-7	-8	-10
L	-2	-2	-2	-3	-4	-5	-6	-7	-8	-10
I	-3	-3	-3	-3	-4	-5	-6	-7	-8	-10
C	-4	-4	-4	-4	-4	-5	-6	-7	-8	-10
A	-5	-5	-5	-5	-5	-5	-6	-7	-8	-10
N	-6	-6	-6	-6	-6	-6	-6	-7	-8	-10

Programación dinámica: algoritmo de Smith-Waterman y alineamientos pareados locales

	C	O	E	L	A	C	A	N	T	H
P	0	0	0	0	0	0	0	0	0	0
E	0	0	0	0	0	0	0	0	0	0
L	0	0	0	0	1	0	0	0	0	0
I	0	0	0	0	1	0	0	0	0	0
C	0	1	0	0	0	0	2	1	0	0
A	0	0	0	0	0	1	1	3	2	1
N	0	0	0	0	0	0	0	2	4	3

El alineamiento local con el máximo score = 4 es:

ELACAN
ELICAN

Tema 2: Alineamientos pareados y búsqueda de homólogos en bases de datos mediante BLAST

Programación dinámica: algoritmos de Smith-Waterman y Needleman-Wunsh ejercicios.

- 1°. Ir a la página del NCBI y descargar las secuencias de los citocromos C P00001 y P00090, y de las proteínas P13569 y P33593, en formato fasta
<http://www.ncbi.nlm.nih.gov/>
- 2°. Ir a la página del Instituto Pasteur en París y hacer un alineamiento global de los citocromos C P00001 y P00090 usando el programa NEEDLE del paquete EMBOSS
<http://bioweb.pasteur.fr/seqanal/interfaces/needle.html>
- 3°. Correr un alineamiento local con las proteínas P13569 y P33593 usando el programa WATER
<http://bioweb.pasteur.fr/seqanal/interfaces/water.html>

Programación dinámica: algoritmos de Smith-Waterman y Needleman-Wunsh ejercicios.

- 2°. Ir a la página del Instituto Pasteur en París y hacer un alineamiento global de los citocromos C P00001 y P00090 usando el programa NEEDLE del paquete EMBOSS
<http://bioweb.pasteur.fr/seqanal/interfaces/needle.html>
- 3°. Correr un alineamiento local con las proteínas P13569 y P33593 usando el programa WATER
<http://bioweb.pasteur.fr/seqanal/interfaces/water.html>

WATER : Smith-Waterman local alignment. (EMBOSS)

Reset Run water your e-mail

(● = required, ● = conditionally required)

Input section

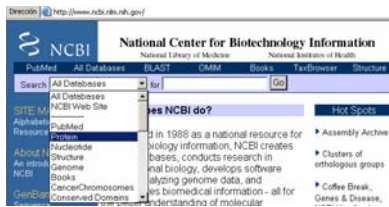
Required section

Advanced section

Output section

Programación dinámica: algoritmos de Smith-Waterman y Needleman-Wunsh ejercicios.

- 1°. Ir a la página del NCBI y descargar las secuencias de los citocromos C P00001 y P00090, y de las proteínas P13569 y P33593, en formato fasta
<http://www.ncbi.nlm.nih.gov/>

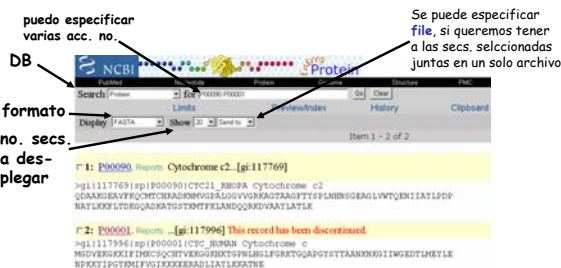


Programación dinámica: algoritmos de Smith-Waterman y Needleman-Wunsh ejercicios.

- alinear a mano los oligonucleótidos TTCATA y TGCTCGTA usando el algoritmo de Needleman y Wunsch y Smith-Waterman el siguiente esquema de ponderación:
match = +5; mismatch = -2; gap = -6

Programación dinámica: algoritmos de Smith-Waterman y Needleman-Wunsh ejercicios.

- 1°. Ir a la página del NCBI y descargar las secuencias de los citocromos C P00001 y P00090, y de las proteínas P13569 y P33593, en formato fasta
<http://www.ncbi.nlm.nih.gov/>



Cuantificación y análisis estadístico de la similitud entre un par de secuencias

Conceptos básicos de teoría de la información

- **INFORMACIÓN** = decremento en el nivel de incertidumbre
- cualitativamente esperamos mayor contenido de información en un **vocabulario rico** que en uno pobre y en **respuestas sorprendentes** que esperadas. Por tanto la información o sorpresividad de una respuesta es inv. prop. a su probabilidad
- cuantitativamente la **información (H)** asociada a un valor de **probabilidad (p)** viene expresada por la siguiente expresión:
$$H(p) = \log_2 1/p = -\log_2 p$$
- valores convertidos a \log_2 se les asigna la unidad **bit** (binary digit), mientras que los que son convertidos a log en base e tienen por unidad los **nats** (natural digits).
- Se describe frecuentemente a la información como un **mensaje de símbolos** emitido por una **fente**. Los símbolos presentan una **distribución de frecuencia**
- Si dicha distribución es plana y existen n símbolos, la p para cada símbolo es $1/n$. La información de cada uno de estos símbolos es su **entropía** = $\log_2 (1/n)$

Cuantificación y análisis estadístico de la similitud entre un par de secuencias

Conceptos básicos de teoría de la información

- Si la distribución de frecuencias no es equiprobable, para calcular la entropía de cada símbolo hay que ponderarla por su p de ocurrencia.

$$H = - \sum_i p_i \log_2 p_i \quad \text{Índice de entropía de Shannon}$$

Ej. 1: para una moneda estándar su entropía es de 1 bit

$$- ((0.5)(-1) + (0.5)(-1)) = 1 \text{ bit}$$

Ej. 2: para una moneda trucada en la que p águila es de 0.75 su entropía es de 0.51 bits

$$- ((0.75)(-0.415) + (0.25)(-2)) = 0.81 \text{ bits}$$

Ej. 3: La entropía de una fuente aleatoria de secuencia de DNA es de 2 bits

$$- ((0.25)(-2) + (0.25)(-2) + (0.25)(-2) + (0.25)(-2)) = 2 \text{ bits}$$

Ej. 4: una fuente de DNA que emite 90% de A ó T y 10% de G ó C es de 1.47 bits

$$- (2(0.45)(-1.15) + 2(0.05)(-4.32)) = 1.47 \text{ bits}$$

Similitud entre pares de secuencias de AA

- El alineamiento de aa difiere del de nt en dos aspectos fundamentales:

1.- Existen más "símbolos" en el alineamiento de aa (20) que de nt (4)

2.- El alineamiento no consiste simplemente en alinear residuos de tal manera que la mayor cantidad coincida, ya que hay que considerar los posibles caminos mutacionales mediante los cuales un aa es sustituido por otro

Cys (UGU) → Tyr (UAU) 1 subst. en la 2a. pos del codón

Cys (UGU) → Met (AUG) 3 subst. Una en cada posición del codón

Por lo tanto alinear Cys con Tyr es 3 veces menos costoso que alinearla con Met

- En el alineamiento de nt generalmente se valora un "match" como +1 y un "mismatch" como -3 (en NCBI BLAST), o como +5/-4 en WU-BLAST, es decir, los nt se consideran idénticos o distintos. Esto, unido a las penalizaciones de gap, define el costo de un alineamiento de nt

- Los alineamientos de proteínas se basan generalmente en una **matriz empírica de costo de sustitución**, derivada de la comparación de secuencias alineadas. Estas matrices empíricas reflejan hasta un cierto punto los caminos mutacionales.

Cuantificación y análisis estadístico de la similitud entre un par de secuencias

- Un script de Perl que calcula la entropía de un archivo usando el índice de Shannon

```
#!/usr/bin/perl -w
use strict;

# Calculadora de entropía de Shannon
my %Count;
my $total = 0;

while (<>) {
    foreach my $char (split(/./, $_)) {
        $Count{$char}++;
        $total++;
    }
}

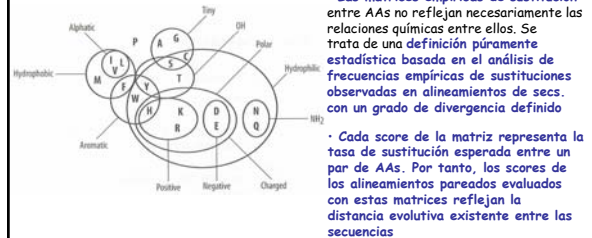
my $H = 0;
foreach my $char (keys %Count) {
    my $p = $Count{$char}/$total;
    $H += $p * log($p);
}

$H = -$H/log(2);

print "H = $H bits \n";
```

Explica lo que hace este script

Similitud entre pares de secuencias de AA



Segmento de un aln. múltiple de citocromos C de primates

Cuantificación y análisis estadístico de la similitud entre un par de secuencias

- Un script de Perl que calcula la entropía de un archivo

```
#!/usr/bin/perl -w

# uso: perl miscript.pl <nombrearchivo>; ó ./miscript <nombrearchivo>

use strict; # llamada al módulo strict

# Calculadora de entropía de Shannon
my %Count; # declaramos un hash que almacenará las cuentas de cada símbolo
my $total = 0; # inicializamos un contador de símbolos totales

while (<>){ # leemos líneas del archivo de entrada
    foreach my $char (split(/./, $_)) { # hacemos un "split" de la línea
        $Count{$char}++; # autoincrementamos el valor de cada llave
        $total++; # autoincrementamos el contador
    }
}

my $H = 0; # inicializamos la variable H (entropía)
foreach my $char (keys %Count){ # iteramos sobre el hash de caracteres
    my $p = $Count{$char}/$total; # probabilidad de cada carácter o símbolo
    $H += $p * log($p); # p * log(p)
}

$H = -$H/log(2); # suma negativa, convertimos base "e" a base 2

print "H = $H bits \n"; # salida
```

Similitud entre pares de secuencias de AA

- Matrices de sustitución de AAs log-odds scores

$$s(a,b) = \frac{1}{\lambda} \log \frac{p_{ab}}{f_a f_b}$$

$s(a,b)$ = score del par a, b

p_{ab} = verosimilitud de la hipótesis a testar; **frecuencia esperada** o **diana**, probabilidad con la que esperamos encontrar a y b apareados en un alineamiento múltiple

$f_a f_b$ = verosimilitud de la hipótesis nula; **frecuencia de fondo**, probabilidad con la que esperamos encontrar a y b en cualquier proteína. Refleja su abundancia o frecuencia

$\lambda = 0.347$ para BLOSUM62. Factor de escalamiento para poder redondear los scores de la matriz a números enteros

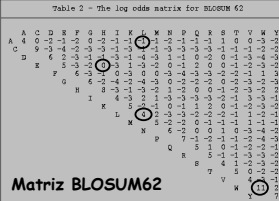
Table 2 - The log odds matrix for BLOSUM 62

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

Matriz BLOSUM62

Similitud entre pares de secuencias de AA

Table 2 - The log odds matrix for BLOSUM 62



Matriz BLOSUM62

- Las matrices empíricas de sustitución entre AAs no reflejan necesariamente las relaciones químicas entre ellos. Se trata de una definición puramente estadística basada en el análisis de frecuencias empíricas de sustituciones observadas en alineamientos de secs. con un grado de divergencia definido

$$s(a,b) = \frac{1}{\lambda} \log \frac{p_{ab}}{f_a f_b}$$

- ¿Porqué difieren los valores entre diferentes sust. conservativas, por ej. L/L y W/W?

$p_{LL} = 0.0371$, $p_{WW} = 0.0065$ Las frecuencias de fondo juegan un papel muy importante. Cuanto más raro es un AA, menos frecuente será que se encuentre apareado consigo mismo por azar

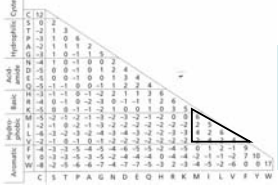
$f_L = 0.099$, $f_W = 0.013$

- ¿Porqué se castiga un apareamiento A/L (chico/alifático) con respecto a uno K/E (+/-)?

$p_{AL} = 0.0044$, $f_L = 0.099$, $f_A = 0.074$
 $p_{KW} = 0.0041$, $f_K = 0.058$, $f_E = 0.054$

Alineamiento pareado de proteínas: matrices de costo PAM

Derivación de una matriz de costo de sustitución PAM250 de M. O. Dayhoff (1978)



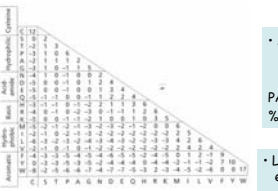
- Para producir una matriz apropiada para estimar similitud entre proteínas más divergentes se toman potencias de la matriz de sustitución PAM 1
- El nivel PAM250, correspondiente a un nivel de identidad global del 20%, es el nivel de divergencia máximo para el que cabe esperar obtener un alineamiento plausible basado únicamente en el análisis de similitud entre las secuencias.
- La matriz da la relación de la frecuencia en la que los pares de aas son observados en comparaciones pareadas de proteínas existentes en bases de datos con respecto a aquellas esperadas por azar, expresadas como "log odds" (ver siguiente página). Lo as intercambiados frecuentemente tienen una puntuación positiva, y aquellos que raramente reemplazan a otros tienen puntuación negativa. Nótese que los reemplazos ocurren más frecuentemente entre aas de propiedades físico-químicas similares (ver como ejemplo los valores en el triángulo)

Similitud entre pares de secuencias de AA

- Matrices de sustitución de AAs: ¿de dónde vienen los log-odds scores?
- La frecuencia diana p_{ab} para un par de AAs corresponde a la probabilidad esperada de encontrar a , b alineados en un alineamiento de secuencias homólogas
- Para estimar la frecuencia diana p_{ab} de un par de AAs con la mayor precisión posible hay que analizar muchos alineamientos pareados entre nuestra secuencia diana y homólogos relacionados con ella a distintos niveles de divergencia evolutiva o distancia genética y calcular la frecuencia a la que ocurre cada par de residuos
- Cuanto más sepamos sobre la biología del par de secuencias alineadas, mejor podremos adecuar la estima de su frecuencia diana. Así p. ej. si alineamos prots de membrana, sus dominios transmembranales tendrán un fuerte sesgo hacia AAs hidrofóbicos, mientras que sus dominios extramembranales tendrán una mayor frecuencia relativa de AAs hidrofílicos. Se trata por tanto claramente de estimas empíricas y adecuadas sólo al caso analizado
- La distancia evolutiva entre las secuencias a analizar es una de las fuentes de información biológica más importantes para hacer una estima adecuada de p_{ab} . Las frecuencias diana dependen fuertemente de la distancia evolutiva entre los pares de secs. analizadas. Si divergieron recientemente, las frecuencias diana deben de ser ajustadas principalmente en base a residuos idénticos. Cuanto más divergentes, la distribución de frecuencias diana debe de ser más plana. Por lo tanto las frecuencias diana se calculan en base a sets de aln. pareados confiables con distinto grado de divergencia. Se obtienen series de matrices correspondientes a estos distintos sets de alineamientos

Alineamiento pareado de proteínas: matrices de costo PAM

Derivación de una matriz de costo de sustitución PAM250 de M. O. Dayhoff (1978)



- La existencia de reversiones (directas o indirectas) produce un ralentizamiento en tasas de sustitución proporcional al grado de divergencia entre secuencias
- Así la relación entre PAM score y % de identidad de secuencia es:

PAM	0	30	80	110	200	250
% identidad	100	75	50	40	25	20

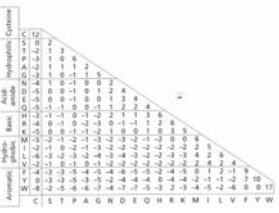
- Los valores de las tablas PAM vienen expresados así:

$$\text{Valor mutación } i \leftrightarrow j = \log \frac{\text{tasa observada } i \leftrightarrow j}{\text{tasa esperada en base a la freq. de aa}}$$

- Este valor se multiplica X10 para evitar decimales

Alineamiento pareado de proteínas: matrices de sustitución PAM

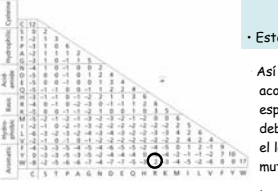
Derivación de una matriz de costo de sustitución PAM250 de M. O. Dayhoff (1978)



- Una medida de divergencia de secuencias de aa es PAM: 1 PAM = 1 Percent Accepted Mutation (1 sustitución/100 residuos)
- Por tanto 2 secuencias que divergen en 1 PAM presentan un 99% de identidad
- Secuencias que divergen en sólo 1% de sus residuos probablemente no hayan sufrido más que una sustitución/síto
- Haciendo una recopilación de sustituciones entre secuencias con 1 PAM de divergencia, y corrigiendo para las abundancias relativas de los aa, se puede derivar una matriz de sustitución PAM1

Alineamiento pareado de proteínas: matrices de costo PAM

Derivación de una matriz de costo de sustitución PAM250 de M. O. Dayhoff (1978)



- Los valores de las tablas PAM vienen expresados así:

$$\text{Valor mutación } i \leftrightarrow j = \log \frac{\text{tasa observada } i \leftrightarrow j}{\text{tasa esperada en base a la freq. de aa}}$$

- Este valor se multiplica X10 para evitar decimales
- Así un valor +2 (p.ej. W ↔ R) implica que la mutación acontece 1.6 veces más frecuentemente que lo esperado por azar. El valor +2 corresponde a 0.2 debido al factor de escalamiento. El valor 0.2 es el log10 del valor de expectación relativa de la mutación. Así el valor de expectación es $10^{0.2} \approx 1.6$
- La probabilidad de dos eventos mutacionales independientes es el producto de sus probabilidades. Al usar logs, se tienen puntuaciones (scores) que se suman en vez de ser multiplicadas, lo que es una ventaja desde la perspectiva computacional

Alineamiento pareado de proteínas: matrices de costo BLOSUM

Matrices BLOSUM de sustitución de aa

Henikoff, S., Henikoff, J. G., and Pietrokovski, S. 1999. Blocks+: a non-redundant database of protein alignment blocks derived from multiple compilations. *Bioinformatics* 15: 471-479.

- Desarrollada por S. Henikoff y J. G. Henikoff para obtener una matriz más robusta que las PAM en la identificación de homólogos distantes, particularmente cuando contienen una proporción significativa de aas hidrofóbicos
- Las matrices **BLOSUM** están basadas en la base de datos **BLOCKS+** de proteínas alineadas; **BLOCKS** Substitution Matrix (<http://blocks.fhcrc.org>)
- Las series de matrices **BLOSUM** se derivaron de alineamientos sin índices (**BLOCKS**) de proteínas considerando sólo pares de alineamientos que no divergieran más de un umbral determinado, por ej. un mínimo de 62 % de identidad, para calcular las frecuencias diana o esperadas de la matriz **BLOSUM62**. Para estos alns. se calcula la razón entre el número de pares de aa observados en cada posición y el número de pares esperados de las frecuencias globales de los aas, expresando los resultados como $\log_{10} X \lambda$.
- Para evitar sesgos en las matrices por sobrerepresentación de secuencias muy similares, se reemplazaron aquellas con similitud > a un umbral dado por un solo representante o por un promedio ponderado (BLOCKS+).
- La matriz **BLOSUM62** es la actualmente favorecida para la mayoría de las aplicaciones por su buen rendimiento empírico y ha reemplazado a las matrices de Dayhoff (**PAM**)

Alineamiento de proteínas: selección de matrices de ponderación - consejos prácticos para la identificación de homólogos

- Clasificación de familias de proteínas atendiendo a su nivel de antigüedad evolutiva I

1. Proteínas antiguas

A) "primeras ediciones": básicamente enzimas del metabolismo central y proteínas involucradas en los procesos de procesamiento de la información genética
Ej. trifosfato isomerasa (TPI), glutamato deshidrogenasa, aminoacyl-tRNA sintetasas, proteínas ribosomales ...

B) "segundas ediciones": homólogos en eucariotes y procariontes, pero con funciones diferenciadas.
Ej: glutatión reductasa humana y la reductasa de Hg de *Pseudomonas* (31% I a lo largo de 438 aa, ($E \times 10^{-32}$))

2. Proteínas de la "edad media"

homólogos en eucariotes pero ausentes en procariontes. Ej: actina humana y la de levadura 88% de I a lo largo de 375 aa, ($E \times 10^{-145}$); otras actinas de levaduras sólo 26% de I a lo largo de 489 aa, ($E \times 10^{-14}$)

Alineamiento de proteínas: selección de matrices de ponderación - consejos prácticos

- Las matrices **PAM** fueron derivadas de las secuencias de proteínas disponibles a finales de los 60s y ppios. de los 70s. Era una base de datos muy reducida y estaba sesgada a proteínas chicas, globulares e hidrofílicas! Al carecer de suficientes homólogos con diversos niveles de divergencia evolutiva tuvieron que emplear supuestos teóricos (caminos mutacionales ...) para inferir las matrices de sustitución para prots. más distantes
- las matrices **PAM** son una pobre elección para alinear (o buscar en las bases de datos) proteínas con dominios hidrofóbicos (p. ej. dominios transmembrana)
- Qué matriz escoger en función del nivel de divergencia esperada (potencial de mira retrospectiva en tiempo evolutivo)

% identidad	PAM	BLOSUM	mira retrospectiva en tiempo evolutivo
20- 50 %	250	45	homólogos en la zona de penumbra
50- 75 %	250	62	ortólogos y parálogos en superfamilias ¹
75- 90 %	160	80	ortólogos y parálogos en familias ²
90- 99 %	40	90	ortólogos muy cercanos

¹Superfamilias de proteínas contienen diversas familias de proteínas con $\geq 30\%$ identidad entre ellas

²Familias de proteínas contienen secuencias con $\geq 85\%$ identidad entre ellas

Estas definiciones fueron acuñadas por Dayhoff et al. (1978)

Alineamiento de proteínas: selección de matrices de ponderación - consejos prácticos para la identificación de homólogos

- Clasificación de familias de proteínas atendiendo a su nivel de antigüedad evolutiva (II)

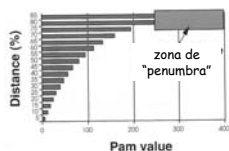
3. Proteínas "modernas"

A) de invención reciente: presentes en plantas o animales pero no en los dos reinos. No presentes en procariontes. Ej. colágeno

B) de invención muy reciente. Por ej. proteínas presentes sólo en vertebrados, tal como la albúmina del plasma sanguíneo

C) mosaicos recientes: proteínas modernas resultantes del barajado de exones (exon-shuffling) como el receptor de LDL o activador de plasminógeno

Alineamiento de proteínas: selección de matrices de ponderación - consejos prácticos para la identificación de homólogos



- A medida que el nivel de divergencia entre pares de proteínas alcanza el valor de PAM250 (~ 20% identidad), comienza a ser dudosa su relación de homología, pudiendo tratarse de secuencias que presentan cierto grado de similitud por azar, en base a composiciones de AAs similares en ambas secuencias !!!
- Al entrar en esta zona de penumbra, es esencial considerar información adicional, particularmente motivos estructurales, para validar o descartar una posible relación de homología

Table 8-1 Observed versus evolutionary distance (expressed as PAM) (assumed point mutation per 100 amino acids) between proteins

Observed percentage difference	Evolutionary distance in PAMs
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	10
11	11
12	12
13	13
14	14
15	15
16	16
17	17
18	18
19	19
20	20
21	21
22	22
23	23
24	24
25	25
26	26
27	27
28	28
29	29
30	30
31	31
32	32
33	33
34	34
35	35
36	36
37	37
38	38
39	39
40	40
41	41
42	42
43	43
44	44
45	45
46	46
47	47
48	48
49	49
50	50
51	51
52	52
53	53
54	54
55	55
56	56
57	57
58	58
59	59
60	60
61	61
62	62
63	63
64	64
65	65
66	66
67	67
68	68
69	69
70	70
71	71
72	72
73	73
74	74
75	75
76	76
77	77
78	78
79	79
80	80
81	81
82	82
83	83
84	84
85	85
86	86
87	87
88	88
89	89
90	90
91	91
92	92
93	93
94	94
95	95
96	96
97	97
98	98
99	99
100	100

- A medida que el nivel de divergencia incrementa (distancias PAM) disminuye el número de diferencias observadas, debido a fenómenos de reversión (homoplasia). Por tanto, si no se cuenta con evidencia estructural, el análisis filogenético de proteínas debe restringirse a aquellas con $\geq 20\%$ de identidad. Los alns. tampoco son confiables

Alineamiento de proteínas: selección de matrices de ponderación - consejos prácticos para la identificación de homólogos

- Para identificar homólogos lejanos de genes codificadores de proteínas, **comparar siempre las secuencias de los productos génicos**. Sólo en ellos quedan reflejadas las constricciones evolutivas que les permiten mantener plegamientos y funcionalidades a lo largo de grandes distancias evolutivas. De ahí la importancia de **incorporar análisis estructurales para la determinación de homología entre secuencias distantes**
- Las secuencias homólogas comparten un ancestro común y por tanto un **plegado común**. Dependiendo de la distancia evolutiva y el camino de divergencia, dos o más homólogos pueden compartir muy pocos residuos estrictamente conservados a nivel de la secuencia primaria. Pero, si se ha podido inferir homología significativa entre A y B, entre B y C y entre C y D, entonces A y D tienen que ser también homólogos entre ellos, aún cuando presenten < 20% de identidad

Tema 2: Alineamientos pareados y búsqueda de homólogos en bases de datos mediante BLAST

BLAST: Basic Local Alignment Search Tool

Gish, W, and DJ States 1993. Identification of protein coding regions by database similarity search. *Nature Genetics* 3:266-72.


Washington University in St. Louis
School of Medicine
WU-BLAST
Welcome to the Washington University BLAST Archives
Serving the world community since 1995

Faster at any sensitivity, more sensitive at any speed, the original gapped BLAST with statistics, providing the performance, features and reliability demanded by technical professionals:

- WU-BLAST 2.0 ... setting a higher standard

For licensed users, the latest release is dated [01-Jan-2006] and is free for academic and nonprofit use.
If you're not using WU-BLAST, you don't know what you're missing!

WU-BLAST en línea por ej. en:
<http://www.ebi.ac.uk/blast2/>

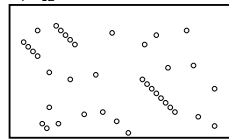


BLAST: Basic Local Alignment Search Tool

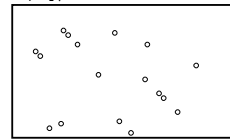
• **Ensemlado**

El valor adecuado de T depende de los valores en la tabla de sustitución empleada, como del balance deseado entre velocidad y sensibilidad. A valores más altos de T , menos palabras son encontradas, reduciendo el espacio de búsqueda. Ello hace las búsquedas más rápidas, a costa de incrementar el riesgo de perder algún alineamiento

$T = 12$



$T = 14$




El tamaño de palabra W es otra variable que controla el número de word hits. $W=1$ producirá más hits que $W=5$. Cuanto más chico sea W más sensible y lenta la búsqueda. La interrelación entre W , T y la matriz de sustitución empleada es crítica, y su selección juiciosa es la mejor manera de controlar el balance entre velocidad y sensibilidad

BLAST: Basic Local Alignment Search Tool

• El algoritmo BLAST

El espacio de búsqueda entre 2 secs. puede ser visualizado como una gráfica con una sec. en cada eje. Sobre esta gráfica podemos visualizar **alineamientos** como una secuencia de letras con o sin gaps.

sec. 2



sec. 1

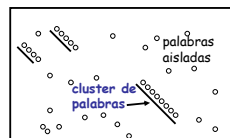
BLAST reporta todos los alns. pareados (HSPs) estadísticamente significativos encontrados en su búsqueda heurística del espacio de búsqueda. Hay que entender que en las búsquedas BLAST siempre hay que hacer un **compromiso entre velocidad y sensibilidad**

El algoritmo heurístico de BLAST sigue tres niveles de reglas para refinar secuencialmente HSPs (High Scoring Pairs) potenciales: **ensemlado**, **extensión** y **evaluación**. Estos pasos conforman una estrategia de refinamiento secuencial que le permite a BLAST explorar todo el espacio de búsqueda sin perder tiempo en regiones de escasa similitud

BLAST: Basic Local Alignment Search Tool

• **Ensemlado**

Las palabras tienden a agruparse en clusters en algunas regiones del espacio. BLAST usa el **two-hit algorithm** para seleccionar regiones con al menos dos palabras agrupadas dentro de una distancia definida sobre la diagonal. De esta manera **se eliminan palabras sin significancia, que carecen de vecinos**. Cuanto más grande la distancia impuesta al algoritmo (Δ), más palabras aisladas serán ignoradas, reduciéndose consecuentemente el espacio de búsqueda



Detalles de implementación:

En **NCBI-BLASTN** las semillas son siempre palabras idénticas. T no es usado. Para hacer BLASTN más rápido se incrementa W por hacerlo más sensible se disminuye W . El valor min. de $W=7$. El algoritmo de two-hit tampoco es usado por BLASTN.

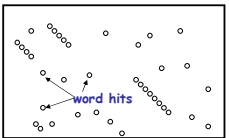
BLASTP (y otros programas basados en aa) usan valores de W de 2 ó 3. Para hacer las búsquedas más rápidas $W=3$ y $T=999$, que elimina todas las palabras vecinas. La distancia del algoritmo two-hit es por defecto = 40 aas. Las palabras que ocurren con una frecuencia significativamente mayor que la esperada por azar corresponden frecuentemente a **regiones de baja complejidad (rbc)** que generalmente son **enmascaradas**. El uso de "soft masking" evita el ensemlado en rbc

BLAST: Basic Local Alignment Search Tool

• **Ensemlado**

BLAST asume que los alineamientos significativos contienen "palabras" en común (serie de letras). BLAST primero determina la localización de todas las palabras comunes ("word hits"). Sólo las regiones que contienen word hits serán usadas como semillas de alineamientos. Así se ahorra mucho espacio a explorar.

sec. 2



sec. 1

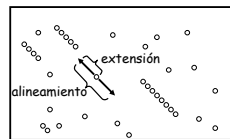
BLAST usa el concepto de **vecindad** para definir un word hit. Esta contiene a la palabra misma y todas las demás cuyo score sea al menos tan grande como T cuando se compara con la matriz de pesado. T corresponde a un threshold (umbral) mínimo de score que han de tener las palabras encontradas. Ajustando el valor del parámetro T y Δ (distancia máxima entre palabras sueltas) se **controla el tamaño de la vecindad**, y por tanto el número de palabras en el espacio de búsqueda

El valor adecuado de T depende de los valores en la tabla de sustitución empleada, como del balance deseado entre velocidad y sensibilidad.

BLAST: Basic Local Alignment Search Tool

• **Extensión**


Una vez que el espacio de búsqueda ha sido ensemlado, pueden generarse alineamientos pareados a partir de semillas individuales. La extensión acontece en ambas direcciones.



En el algoritmo de Smith-Waterman los puntos terminales de un aln. local son determinados después de haber evaluado todo el espacio de búsqueda. BLAST, al ser un algoritmo heurístico, tiene un mecanismo para no tener que explorar todo el espacio de búsqueda y sólo extiende una semilla hasta un determinado punto. Para ello se requiere de una variable X que representa cuanto se permite caer al score del alineamiento después de haber pasado por un máximo. El algoritmo lleva la cuenta de los scores del alineamiento y de caída en base a la matriz de sustitución y de penalización de gaps

Ej. del control de extensión usando +1/-1 para match y mismatch respect., $X=4$, (no gaps)

Pepito Pérez se fue a pescar al lago
Pepito López no vio a Arturo en casa
123456 54345 43 210 1 0 ... <- score aln.
000000 12321 23 456 5 6 ... <- score de caída



longitud de la extensión

Tema 2: Alineamientos pareados y búsqueda de homólogos en bases de datos mediante BLAST

BLAST: Basic Local Alignment Search Tool

- Evaluación**

Una vez extendidas las semillas, los **alns.** resultantes son evaluados para determinar si son o no **estadísticamente significativos**. Los que lo son se denominan **HSPs** (high scoring pairs)

Determinar la significancia de múltiples HSPs no es tan sencillo como sumar los scores de todos los alns. ya que muchos corresponden a extensiones de palabras fortuitas, por lo que no todos los grupos de HSPs tienen sentido. Se define así un **umbral de alineamiento** (aln. threshold T), basado en los scores de los alns. y por lo tanto no considera el tamaño de la base de datos (BD). Cuanto más alto, menos alns. son considerados (Figs. A y B).

Idealmente la relación entre los HSPs debería de ser lo más parecida posible a alns. sin gaps globales, es decir, seguir las diagonales por la mayor distancia posible y no solaparse.

Grupos de HSPs que se comportan de esta manera se denominan **grupos consistentes de HSPs** (Fig. C). Para identificarlos el algoritmo determina las coordenadas de todos los HSPs para cuantificar el solape. Este cálculo es cuadrático. Una vez organizados en grupos consistentes, se calcula un **"final threshold"** para cada grupo que considera todo el espacio de búsqueda (tamaño de la BD). **BLAST reporta todos los que están por encima del E value de corte**

BLAST: Basic Local Alignment Search Tool

- Anatomía de un reporte de NCBI-BLAST estándar**

4. **Alineamientos.** Representan la parte más voluminosa del reporte. Se describen más adelante con detalle

```
>gi|475235461|ref|NP_999401.1| myoglobin [Sus scrofa]
gi|1327688|sp|P02189|MYG_PIG myoglobin
gi|164547|gb|AA031073.1| myoglobin
Length=154

Score = 296 bits (750), Expect = 5e-80
Identities = 144/154 (93%), Positives = 148/154 (96%), Gaps = 0/154 (0%)

Query 1  MGLSDGEWQLVLNVGKVRADI PGRGQEVLRLEFGHPTLEKFKFKHLKSEDEMKASE 60
Sbjct 1  MGLSDGEWQLVLNVGKVRAD+ GRGQEVLRLEFGHPTLEKFKFKHLKSEDEMKASE 60
Query 61  DLKKNATVLTALGGILKKKGHHEAEIKPLAQSHATKHKI PVKYLEFISECIIQVLQSKH 120
Sbjct 61  DLKKNATVLTALGGILKKKGHHEAE+ PLAQSHATKHKI PVKYLEFISECIIQVLQSKH 120
Query 121  PGDFGADAGQAM+KALELFR DMA+ YKELGFGG 154
Sbjct 121  PGDFGADAGQAMKALELFRDMAAKYKELGFGG 154
```

BLAST: Basic Local Alignment Search Tool

- Anatomía de un reporte de NCBI-BLAST estándar**

1.- **Encabezado.** Indica el programa de BLAST y su versión, con la fecha

Request ID

Indica la BD sobre la que se hizo la búsqueda, junto con el no. de secs contenida en ella y el no. de caracteres

Indica cual fue la query y su longitud

2.- **Resumen gráfico de distribución de hits con respecto a la query.**

escala de color que indica el score de los HSPs

Las barras indican la distribución de los HSPs (coordenadas) con respecto

BLAST: Basic Local Alignment Search Tool

- Anatomía de un reporte de NCBI-BLAST estándar**

5. **Pie de página.** Reporta los parámetros de búsqueda y varios estadísticos. Los más importantes son: **DB**, **T**, **E** y la **matriz de sustitución** o **scoring scheme** y **gap penalties** empleados

Database: All non-redundant GenBank CDS translations+PDB+SwissProt+PIR+DBF excluding environmental samples
 Posted date: Mar 6, 2006 5:22 AM
 Number of letters in database: 327,455,400
 Number of sequences in database: 872,833
 Lambda K H
 0.316 0.135 0.398
 Gapped
 Lambda K H
 0.267 0.140 0.140
 Matrix: BLOSUM62
 Gap Penalties: Existence: 11, Extension: 1
 Number of sequences: 872833
 Number of hits to DB: 3803460
 Number of extensions: 145241
 Number of successful extensions: 500
 Number of sequences better than 10: 117
 Number of HSP's better than 10 without gapping: 0
 Number of HSP's gapped: 444
 Number of HSP's successfully gapped: 121
 Length of query: 154
 Length of database: 327455400
 Length adjustment: 111
 Effective length of query: 43
 Effective length of database: 327455400
 Effective search space: 1488882200
 Effective search space used: 891450029
 T: 11
 A: 40
 X1: 16 (7.5 bits)
 X2: 38 (14.6 bits)
 X3: 64 (24.7 bits)
 S1: 41 (20.4 bits)
 S2: 66 (30.0 bits)

$E = kmne^{-\lambda S}$

matriz de sustitución
 gap penalties
 E value umbral usado
 neighborhood word threshold score
 two-hit distance

BLAST: Basic Local Alignment Search Tool

- Anatomía de un reporte de NCBI-BLAST estándar**

3. **Resúmenes de 1 línea.** Indican el nombre de la sec. junto con el score más alto y E value más bajo encontrado para un HSP o grupo de HSPs

Related Structures

Sequences producing significant alignments:	Score	E	Value
gi 48854771 ref NP_053559.1 myoglobin [Homo sapiens] >gi 4495...	316	4e-86	
gi 62511907 gb AA034516.1 myoglobin transcript variant 1 [Homo...	315	1e-85	
gi 386872 gb AA03535.1 myoglobin	315	1e-85	
gi 127361 ref U17140.8 myoglobin	312	4e-85	
gi 127683 sp P02145 MYG_PANTR myoglobin	312	9e-85	
gi 51317414 sp P62735 MYG_WTAE myoglobin >gi 51317413 sp P62734	311	1e-84	
gi 127684 sp P02147 MYG_GORHE myoglobin	311	2e-84	
gi 127360 ref U17140.8 myoglobin	311	2e-84	
gi 15728442 emb CA892020.1 Hypothetical protein [Pongo pygmaeus]	310	5e-84	
gi 1218631 cds U08111 Myoglobin Mutant With Type 45 Replaced By...	309	6e-84	
gi 127689 sp P02148 MYG_ROMEX Myoglobin >gi 1229570 ref U1761377A	308	2e-83	
gi 62501707 sp P68086 MYG_ROMEX Myoglobin >gi 62901706 sp P68...	308	4e-81	

Gene Info
 Structures

BLAST: Basic Local Alignment Search Tool

- Anatomía de un reporte de NCBI-BLAST estándar**

6. **Cladogramas o árboles de NJ o ME.** Navegar por los hits en forma de árboles

Tema 2: Alineamientos pareados y búsqueda de homólogos en bases de datos mediante BLAST

BLAST: Basic Local Alignment Search Tool

RESUMEN de gapped-BLAST

- BLAST es un programa para búsqueda de secuencias similares a una problema en bases de datos. BLAST puede ser usado en línea o localmente.
- Existen **diversos programas BLAST** para comparar todas las combinaciones posibles de secs. problema (aa y nt) con nt o aa DBs. (BLASTN, BLASTP, BLASTX, TBLASTN, TBLASTX) además de variantes de éstos que buscan similitudes en diversas DBs
- BLAST es una **versión heurística del algoritmo de Smith-Waterman** que encuentra matches locales cortos (**palabras**) que intenta extender en forma de alineamientos pareados
- El nuevo algoritmo **gapped-BLAST** requiere al menos de dos palabras o hits no solapados con un score de al menos **7**, ubicados a una distancia máxima **A** el uno del otro, para invocar una extensión del segundo hit. Si el **HSP** generado tiene un score normalizado con un valor de al menos **5g** (**normalized gapped score**) bits, se dispara una extensión con gap
- BLAST reporta además información relativa a la significancia estadística de los HSPs encontrados. El estadístico fundamental es el **valor de expectancia E(E-value)**, que indica la tasa de falsos positivos que cabe encontrar, dada la longitud de la secuencia problema, el tamaño de la base de datos explorada, y el score normalizado del HSP, tal y como indica la **ecuación de Karlin-Altschul**

$$E = kmne^{-\lambda S}$$

- Si bien no existe una teoría estadística para evaluar explícitamente la significancia de alns. con gaps, simulaciones in silico

Identificación de homólogos lejanos mediante PSI-BLAST



Identificación de homólogos lejanos mediante PSI-BLAST

La búsqueda de secuencias distantes en bases de datos mediante **matrices de ponderación sitio específicas** (también conocidas como **perfiles** o **motivos**) son generalmente más adecuadas para la identificación de homólogos con bajo nivel de identidad que el BLASTP estándar

PSI-BLAST (Position-Specific Iterated BLAST) es una modificación de BLASTP que permite la búsqueda de homólogos mediante **perfiles generados automáticamente a partir de alineamientos múltiples derivados de los HSPs encontrados por BLASTP**.

Pasos que sigue el algoritmo de PSI-BLAST

- Búsqueda de homólogos de una sec. problema mediante BLASTP
- Construcción de un aln. múltiple a partir de los HSPs y construcción de un perfil
- El programa compara el perfil construido con la base de datos
- PSI-BLAST determina la significancia estadística de los alns. locales encontrados
- PSI-BLAST puede repetir o iterar los pasos a partir del 2. para construir perfiles cada vez más específicos con las secuencias nuevas encontradas en cada iteración hasta llegar a la convergencia

Identificación de homólogos lejanos mediante PSI-BLAST

Sequences producing significant alignments:	Score (bits)	E Value
gi18438974 ref NP_470948.1 acid tolerance and virulence pro...	796	0.0
gi12454554 gb AA052235.1 Acva precursor [Rhizobium tropici]	475	2e-122
gi117741037 gb AA041509.1 agrobacterium chromosomal virulent...	451	2e-122
gi10954491 gb AA013407.1 AcvB=virulence gene acvB product [Ag...	451	2e-122
gi103044051 gb AA021452.1 acid virulence protein B [Stenotrophob...	425	2e-120
gi1416231 gb J18A03295.1 putative membrane protein [Agrobacter...	373	3e-101
gi10925161 ref J12042288 virulence gene	372	3e-101
gi180377327 gb AA021452.1 virulence protein [Rhodospirillum ...	326	3e-89
gi150741414 emb CA645764.1 CONSERVED HYPOTHETICAL PROTEIN [S...	327	7e-61
gi109551461 ref NP_059798.1 virB [Agrobacterium tumefaciens]...	322	2e-56
gi18138980 ref NP_465465.1 type IV secretory pathway VirD com...	322	3e-29
gi18249498 ref NP_00802848.1 similar to Type IV secretory p...	322	2e-26
gi178691643 ref NP_00816488.1 similar to Type IV secretory p...	322	2e-26
gi178690946 ref NP_00817721.1 similar to Type IV secretory p...	322	4e-26
gi148944001 ref NP_00816811.1 similar to Type IV secretory p...	320	9e-23
gi103181841 gb AA021452.1 VirD [Agrobacterium tumefaciens]	319	0.002

Identificación de homólogos lejanos mediante PSI-BLAST

matrices de ponderación sitio específicas (Position Specific Scoring Matrices PSSMs)

Se construyen usando algoritmos de cadenas ocultas de Markov (HMMs). En esencia, para un alineamiento múltiple se consideran tanto las posiciones como las frecuencias de los estados de carácter observados para cada sitio. Residuos que conservados en una determinada posición reciben un score positivo muy alto, mientras que los raros en dicha posición reciben un score alto negativo. Residuos que ocupan posiciones muy variables reciben scores próximos a cero.

12345678910	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
1 Y	-2	-2	-2	-3	-2	-1	-2	-3	2	-1	-1	-2	-1	3	-3	-2	-2	7	-1	-1
2 L	-1	-2	-3	-4	-1	-2	-3	-4	-3	2	4	-2	2	0	-3	-2	-1	-2	1	-1
3 P	-1	-2	-2	-3	-2	-1	-2	-2	-3	-3	-1	-3	-4	8	-1	-1	-4	-3	-3	-3
4 S	1	-1	0	-1	0	0	-1	-1	-3	-3	0	-2	-3	-1	5	1	3	-2	-2	-2
5 C	-1	-4	-3	-4	9	-3	-4	-3	-2	-2	-3	-2	-3	-1	-1	-3	-3	-1	-1	-1
6 T	0	-1	0	-1	-1	-1	-1	-2	-2	-3	-1	-2	-3	-1	4	3	-2	-2	-2	-2
7 V	-2	-3	-3	-4	-2	-3	-4	1	-1	-1	-3	-1	5	-4	-2	-2	1	7	-2	-2
8 Y	-1	-1	-1	-2	0	-1	-2	6	-2	-1	-1	-1	-1	-1	-1	0	5	-2	-2	-2
9 V	-1	-2	-2	-2	-1	-2	-2	-2	1	2	-2	0	-1	-2	-2	-2	-1	-2	-1	-4
10 S	-1	-1	-1	-3	3	3	-2	-1	-2	1	0	-1	-2	-2	2	-1	-3	-2	-2	-2

Ejemplo de una PSSM calculada para los 10 primeros residuos de un alineamiento múltiple de proteínas HoxA de eucariotes. Sólo se muestra una pequeña parte de las secuencias que incluidas en el alineamiento múltiple.

Ejemplo de una PSSM calculada para los 10 primeros residuos de un alineamiento múltiple de proteínas HoxA de eucariontes. Sólo se muestra una pequeña parte de las secuencias que incluídas en el alineamiento múltiple

Identificación de homólogos lejanos mediante PSI-BLAST

Sequences producing significant alignments:	Score (bits)	E Value
gi148944001 ref NP_00816811.1 similar to Type IV secretory p...	322	3e-71
gi10328111 emb CA645764.1 HYPOTHETICAL TRANSMEMBRANE PROTEIN...	241	5e-61
gi187133284 gb AA024004.1 Type IV secretory pathway VirD com...	309	2e-22
gi184705067 ref NP_01018567.1 virulence protein [Parvularcul...	87.8	2e-14
gi103044051 gb AA021452.1 acid virulence protein B [Stenotrophob...	425	2e-104
gi103181841 gb AA021452.1 VirD [Agrobacterium tumefaciens]	319	0.002
gi171132491 gb AA024004.1 Type IV secretory pathway VirD com...	309	4e-20
gi184705067 ref NP_01018567.1 virulence protein [Parvularcul...	87.8	2e-14
gi103044051 gb AA021452.1 acid virulence protein B [Stenotrophob...	425	2e-104
gi103181841 gb AA021452.1 VirD [Agrobacterium tumefaciens]	319	0.002
gi171132491 gb AA024004.1 Type IV secretory pathway VirD com...	309	4e-20
gi184705067 ref NP_01018567.1 virulence protein [Parvularcul...	87.8	2e-14
gi103044051 gb AA021452.1 acid virulence protein B [Stenotrophob...	425	2e-104
gi103181841 gb AA021452.1 VirD [Agrobacterium tumefaciens]	319	0.002
gi171132491 gb AA024004.1 Type IV secretory pathway VirD com...	309	4e-20
gi184705067 ref NP_01018567.1 virulence protein [Parvularcul...	87.8	2e-14
gi103044051 gb AA021452.1 acid virulence protein B [Stenotrophob...	425	2e-104
gi103181841 gb AA021452.1 VirD [Agrobacterium tumefaciens]	319	0.002
gi171132491 gb AA024004.1 Type IV secretory pathway VirD com...	309	4e-20
gi184705067 ref NP_01018567.1 virulence protein [Parvularcul...	87.8	2e-14
gi103044051 gb AA021452.1 acid virulence protein B [Stenotrophob...	425	2e-104
gi103181841 gb AA021452.1 VirD [Agrobacterium tumefaciens]	319	0.002
gi171132491 gb AA024004.1 Type IV secretory pathway VirD com...	309	4e-20

Tema 2: Alineamientos pareados y búsqueda de homólogos en bases de datos mediante BLAST

Identificación de homólogos lejanos mediante PSI-BLAST

Aspectos a cuidar al calcular PSSMs

- 1.- Hay que tener cuidado de no incluir secuencias no homólogas. Revisar alineamientos pareados, estructura de dominios y no fiarse de las anotaciones. Muchas están mal anotadas !!!

Utilizar:

<http://www.ncbi.nlm.nih.gov/COG/>
<http://psort.hgc.jp/>
<http://www.predictprotein.org/newwebsite/>
http://www.ch.embnet.org/software/TMPRED_form.html
<http://www.expasy.org/>
...

para caracterizar a las proteínas dudosas ...

- 2.- Eliminar regiones de baja complejidad.

Usar SEG y COILS

http://www.ch.embnet.org/software/COILS_form.html

URLs de algunas de las principales bases de datos de secuencias (DNA, Prot.), familias/dominios/motivos de proteínas y estructuras

Blocks and Blocks+ : <http://blocks.fhcrc.org/>
DBJ : <http://www.ddbj.nig.ac.jp/>
EMBL : <http://www.ebi.ac.uk/embl/>
Entrez : <http://www.ncbi.nlm.nih.gov/Entrez/>
GenBank : <http://www.ncbi.nlm.nih.gov/Genbank/>
InterPro : <http://www.ebi.ac.uk/interpro/>
MEDLINE : <http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=PubMed>
PDB : <http://www.rcsb.org/pdb/>
PIR : <http://www.nbrf.georgetown.edu/>
Pfam : <http://www.sanger.ac.uk/Pfam/>
PRINTS : <http://www.bioinf.man.ac.uk/dbbrowser/PRINTS/PRINTS.html>
ProDom : <http://protein.toulouse.inra.fr/prodom.html>
PROSITE : <http://www.expasy.ch/prosite/prosite.html>
SR5 "mother" server : <http://srs.ebi.ac.uk/>
SWISS-PROT and TrEMBL at EBI : <http://www.ebi.ac.uk/swissprot/>

PRÁCTICAS: aprendiendo a usar PSI-BLAST para identificar homólogos lejanos

- 1) Descarga la secuencia Q57997 y haz un análisis de PSI-BLAST. Preguntas:
 - Qué tipo de función podría tener esta proteína?
 - Cuántos homólogos encontraste en la primera búsqueda (BLASTP)?
 - Cuántos ciclos o iteraciones tuviste que correr hasta la convergencia? Cuántos homólogos pescaste?
- 2) Compara estos resultados con el análisis descrito en el tutorial de PSI-BLAST que encontrarás en la página del NCBI bajo:
<http://www.ncbi.nlm.nih.gov/Education/BLASTinfo/psi1.html>
- 3) Ve a la página de nuestro curso y haz los ejercicios propuestos que encontrarás en el directorio Ejercicios/BLAST

Alineamientos múltiples (AM)

- Existen diversos algoritmos (además de matrices de sustitución y de "gap penalty") para la generación de AMs. Unos son **exhaustivos** (garantizan encontrar el alineamiento óptimo) y otros son **heurísticos** (no lo garantizan)
- No existe un algoritmo ideal para todas las situaciones. Para búsquedas en bases de datos se emplean algoritmos heurísticos para encontrar primeramente **alineamientos locales** (FastA y BLAST). Para análisis filogenéticos generalmente preferiremos métodos que produzcan **alineamientos globales**.
- Algoritmos basados en **programación dinámica** (PD) aseguran encontrar la solución óptima o el mejor **alineamiento global** para 2 secuencias. Se trata de un algoritmo $O(N^2)$, ya que el tiempo y memoria que demandan es proporcional al producto de las long. de ambas secuencias ($N1 \times N2$). Se puede generalizar el proceso para la comparación de múltiples secuencias, usando la **función de objetividad** llamada **suma ponderada de pares (WSP)**:

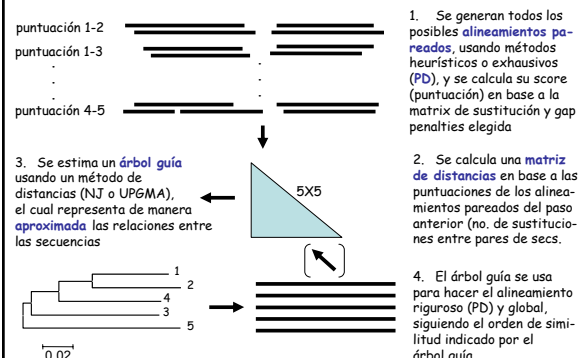
$$\sum \sum W_{ij} D_{ij}$$

Donde D_{ij} es la puntuación de cada posible par de secuencias y W_{ij} es un factor de ponderación. Algoritmos de PD se pueden emplear para encontrar el AM que da el mejor valor posible de la función WSP. El problema radica en que computacionalmente la complejidad crece exponencialmente con cada nueva secuencia que se añade (complejidad $O(N^M)$)

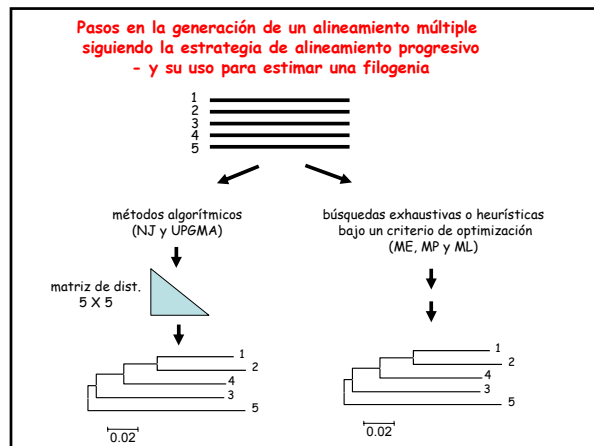
Consejos finales para el uso eficiente de BLAST

0. Antes de iniciar búsquedas con BLAST, hay que **escanear las secs.** para detectar la presencia de múltiples dominios, rep. repetitivas, motivos y péptidos señal usando las herramientas o servidores apropiados (**SMART, PROSITE, PFAM, CDD, PSORT ...**)
1. Para búsquedas de secuencias homólogas distantes **usa AAs y PSI-BLAST** siempre que sea posible.
2. **de PSSMs.** Usa todos los criterios adicionales que consideres relevantes para inferir la homología de manera certera. No te fíes de las anotaciones, las hay erróneas. También conviene ser crítico con las proteínas hipotéticas, puesto que su existencia no se ha demostrado experimentalmente y con frecuencia presentan extremos N terminales más largos que los de las proteínas de verdad (problema de predecir adecuadamente el inicio de traducción).
3. Ajusta el valor de los parámetros de búsqueda de manera adecuada al problema a resolver. **El valor de los parámetros determina lo que puedes encontrar.** Así por ejemplo búsquedas con NCBI-BLASTN con valores por defecto de match (+1) y mismatch (-3) tienen una frecuencia diana de 99% de identidad. No busques genes de humano y nemátodo con NCBI-BLASTN...
4. Haz **controles**, especialmente cuando se trate de similitudes en la **zona de penumbra**. Así por ejemplo puedes hacer un **"barajado"** de la secuencia problema a mano o mejor aún, usando un sencillo script de Perl. Si tras barajar los caracteres de tu secuencia sigues encontrando hits similares en la zona de penumbra el parecido se debe simplemente a un sesgo composicional compartido entre ambas secs. y no a homología

Pasos en la generación de un alineamiento múltiple siguiendo la estrategia de alineamiento progresivo



Tema 2: Alineamientos pareados y búsqueda de homólogos en bases de datos mediante BLAST



Alineamientos múltiples progresivos usando Clustal -un ejemplo: alineamiento de GDPs dependientes de NAD

1.- Selecciona modo de aln y fichero a alinear (en este caso las secs. están escritas en formato FASTA)

```
>gp1yeast
MGAADRLNLTGHLNAGKRSSSVLSAAEFKTVIGSNWOTTTA
KVVAENCKGYEVFAFVQMVFEENGEELTRIINRWKVKYLPIT
LFDNLVNPDLDSVDKVDIVFNIFHQFLRICQLKGVDSHVAISC
LQGFVYAGVOLLSTYTELAIQGLSANTIAEVAQIBETVAT
HIFKFGGKQVDRVKALFLRFYFVYVIEVINGISGQALEXVAL
GGFVGLGWGNNASAIQVGLSEIIFGQMFESSEETTYQESAGVA
DLITTCAGSHNVKARLHATSGHDMCEKELLNGSAGLITCEVHEW
LETQSVDFEFLFANYQIVYNYFIMHLEFELLEDLED
>gp2drome
MADVNVICVSGSMGSAIAKIVGANAALFEERVMFYELIDGKK
LPTINEHENVYKGLHLPFVAVVDFVDAANDLIPVWSPFIFN
FCQLLKGKIKNAISLIRGFKAGGGIDLSHITRIPCVLMGANL
ANEVAGNFRTTIGCTKQYKVLRLDFQNHFRVYVDDADAVCVGA
LQNTVACAGVFGVGLSNTFANVILGLMDIRVDFYFGKLETF
EGCGVADLITTVRSRAFTVSGTKTIELEKMLNQKLGQFPTAEVNY
```

Alineamientos múltiples progresivos usando Clustal

- La familia Clustal es posiblemente la más popular para hacer AMs de nt y aa
- Existen versiones para todas las plataformas y en red (<http://www2.ebi.ac.uk.clustalw>)
- La primera versión (Clustal) salió en 1988, la última, **ClustalX**, en 1997 (última Vers. = 1.81)
- ClustalX (X-windows Clustal)** lee secuencias en diversos formatos, calcula un **árbol guía NJ**, usando algoritmos heurísticos o exhaustivos sobre aln locales basado en **distintas matrices de peso y de penalización de gaps afines y sitio-específicas**. Puede hacer **alineamientos de perfiles** y existen diversas **herramientas de control de calidad del AM** y para hacerlo en base a criterios estructurales, usando por ej. **máscaras estructurales**. Partes del alineamiento o secuencias particulares pueden ser **realineadas** para ir obteniendo un aln global cada vez mejor. Es decir, ClustalX no sólo genera alineamientos (como ClustalW), sino que éstos pueden ser editados y mejorados interactivamente por el usuario. Además, ClustalX (y ClustalW) permite la **reconstrucción y visualización de árboles NJ** y hacer **análisis de bootstrap** sobre los alineamientos. Finalmente, los AMs pueden ser escritos en **diversos formatos de salida** (CLUSTAL, FASTA, NEXUS, PHYLIP ...)

Alineamientos múltiples progresivos usando Clustal -un ejemplo: alineamiento de GDPs dependientes de NAD

Alineamientos múltiples progresivos usando Clustal -aspectos prácticos

- Para obtener un AM con clustal tenemos que tener todas las secuencias homólogas en un solo fichero. Estas secs. pueden estar escritas en diversos formatos (FASTA, EMBL, SWISS-PROT ...)
- Sobre este fichero se puede correr un primer análisis usando las opciones por defecto de Clustal
- Según el grado de divergencia de las secuencias a analizar, puede ser muy recomendable probar distintas series de matrices y valores de gap penalty
- Clustal es adecuado para alinear sets de secuencias totalmente colineares (no usar para ensamblar contigs) y que presenten el mismo orden de dominios estructurales
- Condiciones en las que Clustal no puede operar de manera óptima**
 - Si tenemos unas pocas secuencias muy divergentes de una superfamilia; ajustar "delay parameter" y/o usar modo de alineamiento de perfiles, preferentemente con máscara estructural
 - Sesgo composicional en aas hidrofílicos (G, P, S, N, D, Q, E, K, R) pueden introducir demasiados gaps (penalizaciones de indel sitio-específico)

Alineamientos múltiples progresivos usando Clustal -un ejemplo: alineamiento de GDPs dependientes de NAD

Alineamientos múltiples progresivos usando Clustal
-un ejemplo: alineamiento de GbPs dependientes de NAD

Formatos de secuencias
II) PHYLIP

- **Phylip (interleaved):** no. seqs, no. caracteres
nombre secuencias (máx 10 caracteres) espacio, secuencia ...

```

3      100
R._galegae CCGCUGGUCA CCUCCGGCAA GCGGCGCAUC CACCAGGAAG CGCCUUCUA
M._plurifa ...G.C.A.G ..GU..AGCU ...U..... .CCG..U..GG...
B._japonic ...G.CAAGU .GGAA...CU ..... .GA....

      CGUCGAUCAG UCGACCGAAG GCCAGAUCCU GGUCACCGGC AUCAAGGUCG
U.....C.....G.....CG.....U.....UC
.AC...C... ..C..... CUG.A..U.. C.....

```

- **Phylip (sequential or non-interleaved)**

```

3      100
R._galegae CCGCTGGTCA CCTCCGGCAA GCGCGCCATC CACCAGGAAG CGCCTTCCTA
CGTCGATCAG TCGACCGAAG GCCAGATCCT GGTACCGGC ATCAAGGTCG
M._plurifa CCGTGCAGC CCGTGCAGCT GCGTGCATC CACCAGCGG CTCGGGCTA
TGTCGACCA TCGACCGAAG CGCAGATCCT GGTACCGGC ATCAAGGTC
B._japonic CCGGTCAAGT CGGAGGCCT GCGCGCCATC CACCAGGAAG CGCCGACCTA
CACCGACCA TCGACCGAAG CTGAATTCT GTTACCGGC ATCAAGGTCG

```

Alineamientos múltiples progresivos usando Clustal
-un ejemplo: alineamiento de GbPs dependientes de NAD

Formatos de secuencias
III) NEXUS

```

#NEXUS
[OJO!!!, no usar guiones-, sólo guiones bajos]

[ taxa block ]
BEGIN TAXA;
DIMENSIONS NTAX=3;
TAXLABELS
R._galegae;
M._plurifarium;
B._japonicum;
END;

[ character block ]
BEGIN CHARACTERS;
DIMENSIONS NCHAR=100;
FORMAT DATATYPE=DNA MISSING=? GAP=- MATCHCHAR=. INTERLEAVE=yes ;
MATRIX
[
      10      20      30      40      50]
[
      *      *      *      *      *]
R._galegae CCGCTGGTCACTCCGGCAAGCGCGCCATCCACCAGGAAGCGCCTTCCTA
M._plurifarium ...G.C.A.G..GT..AGCT...T.....CCG..T..GG...
B._japonicum ...G.CAAGT.GGAA...CT......GA....

[
      60      70      80      90      100]
[
      *      *      *      *      *]
R._galegae CGTCGATCAGTCGACCGAAGGCCAGATCTGGTACCGGCATCAAGGTCG
M._plurifarium T.....C.....G.....CG.....T.....TC
B._japonicum .AC...C.....C.....CTG.A..T..C.....
;
END;

```

Formatos de secuencias
I) FASTA

- Existen una gran cantidad de estilos o formatos de presentación de secuencias. Muchos programas de análisis filogenético usan su propio formato (Phylip, Nexus, Mega ...)
- El formato más sencillo es el **FASTA**, en el que cada secuencia se identifica mediante un renglón descriptor que comienza con > en el siguiente renglón comienza la secuencia

```

>R._galegae
CCGCTGGTCACTCCGGCAAGCGCGCCATCCACCAGGAAGCGCCTTCCTA
CGTCGATCAGTCGACCGAAGGCCAGATCTGGTACCGGCATCAAGGTCG

>M._plurifarium
CCGCTGCAGCGCGTGCAGCTGCGTGCCATCCACCAGCGGCTCCGGCCTA
TGTCGACCACTGACCGGAAGCGCAGATCTGTTACCGGCATCAAGGTCG

>B._japonicum
CCGCTCAAGTCCGAAGGCCTGCGGCCATCCACCAGGAAGCGCGCACTA
CACCGACCACTCCACCAGGCTGAATTCCTGTCACCGGCATCAAGGTCG

```

Formatos de secuencias
IV) MEGA

```

#mega
TITLE 3 atpD sequences, 100 nt
Number of Sequences: 3. Sequence Length: 150
Output on Sábado, 19 de Febrero de 2005

#R._galegae
CCGCUUGUACCCUCCGGCAAGCGCGCAUCCACCAGGAAGCGCUUCCUACGUCGUAUCAG
UCGACCGAAGGCCAGAUCCUGUACCCGCAUCCAGGUCGACCUUGCGCCUUAUC
GCAAGGGCGGCAAGAUCCGCUUUCGCG

#M._plurifarium
CCGUGUACGCGGUGGAGUGGUGGCAUCCACCAGCGCGGCGGCUUUGUACGACCAAG
UCGACCGAAGGCCAGAUCCUGUACCCGCAUCCAGGUCGACCUUGCGCGCCUUAU
GCGCGCGGCGGCAAGAUCCGCGUUGCGG

#B._japonicum
CCGUGAAGUGGGAAGGCGUGCGCGCAUCCACCAGGAAGCGCGACCUACCGGACCAAG
UCCACGGAAGCUGAAAUUCUGUACCCGCAUCCAGGUCGUGAUCCUGGCGCCUUAU
GCGAAGGGCGGCAAGAUCCGCGUUGCGG

```