

**Introducción a la Filoinformática:**  
**IBBM-UNLP/CONYCEP, Argentina. 2-6 Julio 2018.**

Pablo Vinuesa (vinuesa@ccg.unam.mx)

Programa de Ingeniería Genómica, CCG-UNAM, México  
<http://www.ccg.unam.mx/~vinuesa/>

Todo el material del curso (presentaciones, tutoriales y datos de secuencias) lo encontrarás en:  
<https://github.com/vinuesa/intro2phyloinfo>

**Tema 2: alineamientos pareados y búsqueda de homólogos en bases de datos**

- 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 búsqueda de secuencias homólogas
- prácticas: uso de NCBI BLAST en línea

**Homología entre secuencias de DNA y proteína:**  
**tipos de mutaciones en secs. codificadoras de proteínas**

secuencia ancestral: pos. codón 123 codones ATG TGT TTT GAT GCA AA M C F D A

especie A: ATG TAT TTT CAT GCA M T F H A (no-sinónima)

especie B: ATG --- TTC GAC GCA M F D A (sinónimas y delección en marco)

especie C: ATG TGT TT- G ATG CAX M C L M X (delección fuera de marco)

- Todas las mutaciones en 2ª posiciones resultan en sustituciones no sinónimas
- 96% de mutaciones en 1ª posiciones resultan en sustituciones no sinónimas
- Casi todas las sustituciones sinónimas ocurren en las 3ª posiciones
- las delecciones 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

**Estadísticos de Karlin-Altschul de similitud entre secuencias:**  
**frecuencias diana, lambda y entropía relativa**

Los atributos más importantes de una matriz de sustitución son sus frecuencias esperadas o diana implícitas para cada par de aa en sus respectivos scores crudos. Estas frecuencias esperadas representan el modelo evolutivo subyacente. Los scores que han sido re-escalados y redondeados (scores representados en la matriz) son los scores crudos  $s_{ab}$ . Para convertirlos a un score normalizado (log-odd score original) tenemos que multiplicarlos por  $\lambda$ , una constante específica para cada matriz.  $\lambda$  es aprox. igual al inverso del factor de escalamiento ( $c$ ).

$$s(a,b) = \frac{1}{\lambda} \log \frac{p_{ab}}{f_a f_b} \quad p_{ab} = f_a f_b e^{\lambda s_{ab}} = \text{score normalizado}$$

por tanto, para despejar  $\lambda$  necesitamos  $f_a f_b$  y encontrar el valor de  $\lambda$  para el que la suma de las frecuencias diana implícitas valga 1.

$$\sum_{a=1}^a \sum_{b=1}^b p_{ab} = \sum_{a=1}^a \sum_{b=1}^b f_a f_b e^{\lambda s_{ab}} = 1$$

Una vez calculada  $\lambda$ , se usa para calcular el valor de expectación (E) de cada HSP (High Scoring Pair) en el reporte de una búsqueda BLAST

Dado que las  $f_a f_b$  de los residuos de algunas proteínas difieren mucho de las frecuencias de residuos empleadas para calcular las matrices PAM y BLOSUM, versiones recientes de BLASTP y PSI-BLAST incorporan una "composition-based  $\lambda$ " que es "hit-específica"

**Protocolo básico para un análisis filogenético de secuencias moleculares**

**Tema 3:** alineamientos pareados, búsquedas de homólogos en bases de datos

Colectación de secuencias homólogas  
 ↓  
 • BLAST y FASTA

Alineamiento múltiple de secuencias  
 ↓  
 • Clustal, T-Coffee, muscle...

Análisis evolutivo del alineamiento y selección del modelo de sustitución más ajustado  
 ↓  
 • tests de saturación, modeltest, ...

Estima filogenética  
 ↓  
 • NJ, ME, MP, ML, Bayes ...

Pruebas de confiabilidad de la topología inferida  
 ↓  
 • proporciones de bootstrap probabilidad posterior ...

Interpretación evolutiva y aplicación de las filogenias

**Homología entre secuencias de DNA y proteína:**  
**alineamiento y tipos de mutaciones**

secuencia ancestral: pos. codón 123 codones ATG TGT TTT GAT GCA AA M C F D A

alineamiento de sitios homólogos para tres secs.  
 especie A: ATG TAT TTT CAT GCA  
 especie B: ATG --- TTC GAC GCA  
 especie C: ATG TGT TT- GAT GCA

cambio de marco de lectura !!! posible pseudogen.

Transiciones (ti) purina - purina  
 $\alpha_{A,C}$   
 $\beta_{A,C}$

Transversiones (tv) pur. <-> pyr.  
 $\beta_{C,G}$   $\beta_{A,T}$   $\beta_{G,T}$

Transiciones (ti) pirimidina - pirimidina  
 $\alpha_{C,G}$

- existen 4 tipos de ti y 8 de tv
- las tasas de sustitución de ti ( $\alpha$ ) son generalmente mucho más altas que las de tv ( $\beta$ )

**Alineamientos pareados y búsqueda de homólogos en bases de datos**

Los alineamientos pareados son la base de los métodos de búsqueda de secuencias homólogas en bases de datos

- Si dos proteínas o genes se parecen mucho a lo largo de toda su longitud asumimos que se trata de proteínas o genes homólogos, es decir, descendientes de un mismo ancestro común (cenastro).
- Por ello una de las técnicas más utilizadas para detectar potenciales homólogos en bases de datos de secuencias se basa en la cuantificación de la similitud entre pares de secuencias y la determinación de la significancia estadística de dicho parecido. Estas magnitudes son las que reportan los estadísticos de BLAST.

```

> g1171548896[ref|PF_00669120.1] Translation elongation factor G:small GTP-binding protein domain
[Nitrosomonas eutropha C71]
g1171486077[gb|E018626.1] 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 MTRFSLKTRNIGIMHIDAGKTTTTERLVYTRIRKIKIGSTHGASQMDMNAQDQERG 60
M++ LE+ FNIGIMAHIDAGKTTTTERL+TTS HE+GE H+GA+ MDM QDQERG
Sbjct 1 MSKRNLKRVFNIGIMHIDAGKTTTTERLFTVGSKLGSEVHGDAATHDMMDQDQERG 60

Query 61 xxxxxxxxxxxxxx-----DHRNIDTFSHVDFTVERSLVLGAVLDAQSGVE 113
ITFSAFT W +HRN+IDTFSHVDFTVERSLVLGDA V + SV+
Sbjct 61 ITFSAFTFWKMGNTYERHNLVITDFSHVDFTVERSLVLGDACTVFCVSVQVQ 120 (... truncado)
    
```



### Estadísticos de Karlin-Altschul para alineamientos locales

Karlin, S., and Altschul, S. F. 1990. Methods for assessing the statistical significance of molecular sequence features by using general scoring schemes. Proc Natl Acad Sci U S A 87: 2264-268.

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

Esta ecuación indica que el número de alineamientos esperados por azar ( $E$ ) durante una búsqueda de similitud en una base de datos de secuencias está en función de: el tamaño del espacio de búsqueda ( $m, n$ ), el score normalizado ( $\lambda S$ ) del HSP y una constante de valor pequeño ( $k$ )

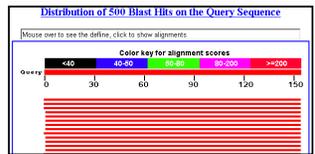
$E$  Describe el ruido de fondo por azar presente en matches de dos secs.

$m$  = número de símbolos en la secuencia problema  
 $n$  = número de símbolos en la base de datos  
 $k \approx 0.1$  constante de ajuste para considerar HSPs altamente correlacionados

### BLAST: Basic Local Alignment Search Tool

• Anatomía de un reporte de NCBI-BLAST estándar

```
BLAST 2.2.13 [Nov 27 2005]
reference:
Altschul, Stephen F., Thomas L. Madden, Alejandro A. Schaffer,
Nguyen-Thao, Zhong Zhang, Webb Miller, and David J. Lipman
(1997). "Gapped BLAST and PSI-BLAST: a new generation of
protein database search programs". Nucleic Acids Res. 25:3389-3402.
DOI: 10.1093/nar/25-12667-9201132765.BLA8P04
Database: All non-redundant GenBank CDS
Translations+PDB+SwissProt+PIR+PDB excluding environmental samples
3,420,754 sequences; 1,167,289,757 total letters
Please refer to the BLAST FAQ
Have any questions?
Query: human_myoglobin
Length:154
```



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 las HSPs

Las barras indican la distribución de los HSPs (coordenadas) con respecto a la secuencia problema (query), indicando en una escala de color el score de los alns. medidos en bits

### BLAST: Basic Local Alignment Search Tool

• Anatomía de un reporte de NCBI-BLAST estándar

4. Alineamientos. Representan la parte más voluminosa del reporte. Además de la información estadística, indica las coordenadas de inicio y fin de las secuencias query y subject. Si la búsqueda involucra secuencias de DNA, también se indica direccionalidad de las hebras Q/S (plus/plus; plus/minus).

```
> q|147523546|eef(INP_999401.1) | M myoglobin [Sus scrofa]
q|127688|ep|E02189|MYG_PTC | M Myoglobin
q|164547|lab|AAAS1073.1 | M myoglobin
Length=154 normalized score
Score = 296 bits (758), Expect = 5e-80
Identities = 144/154 (93%), Positives = 140/154 (96%), Gaps = 0/154 (0%)
Query 1 MGLSDGEMQLVLMVGVKVEADI PGHGQVLRLEFGHPETLEKDFKHLKSDDEMKASE 60
          MGLSDGEMQLVLMVGVKVEAD+ GHGQVLRLEFGHPETLEKDFKHLKSDDEMKASE 60
Sbjct 1 MGLSDGEMQLVLMVGVKVEADVAGHGQVLRLEFGHPETLEKDFKHLKSDDEMKASE 60
Query 61 DLKRGKATVLTALGGLLKRGKHEARLPLAQSHATKHKI PVKYLEP I SECLIVLQSRH 120
          DLKRGK TVLTALGGLLKRGKHEAR+ PLAQSHATKHKI PVKYLEP I E IIVLQSRH 120
Sbjct 61 DLKRGKTVLTALGGLLKRGKHEARLPLAQSHATKHKI PVKYLEP I E IIVLQSRH 120
Query 121 PDGFGADAQGANRKALELFRMDASNYKELGFGQ 154
          PDGFGADAQGANRKALELFR DMA+ YKELGFGQ
Sbjct 121 PDGFGADAQGANRKALELFRMDAARVYKELGFGQ 154
```

### BLAST: Basic Local Alignment Search Tool

BLAST consta de una familia de programas. Los 5 ppales son:

- BLASTN (nt-nt), BLASTP (p-p), BLASTX (translated nt-p),
- TBLASTN (p-translated nt), usado en mapeo de prots contra DNA genómico
- TBLASTX (translated nt - translated nt) usado en la predicción de genes

y variantes de BLASTP como PSI- y PHI-BLAST

### 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 (Bits)	E Value
q 14885477 eef(INP_005359.1) myoglobin [Homo sapiens]  g 14495...	316	6e-86
q 162511907 lab AAAS4516.1  myoglobin transcript variant 1 [Homo sapiens]  g 14495...	315	1e-85
q 127688 ep E02189 MYG_PTC   M Myoglobin  g 14495...	315	4e-85
q 147523546 eef(INP_999401.1) myoglobin [Sus scrofa]  g 14495...	312	9e-85
q 15131714 ep E02189 MYG_PTC   M Myoglobin  g 15131714 ep P62734	311	2e-84
q 112725614 ep E02189 MYG_PTC   M Myoglobin  g 112725614 ep P62734	311	2e-84
q 12292691 p J171160.8  myoglobin  g 12292691 p J171160.8	311	2e-84
q 15172144 emb CA09266.1  hypothetical protein [Pongo pygmaeus]  g 15172144 emb CA09266.1	310	5e-84
q 1210631 pdb 1QMM1   Myoglobin Mutant With Lys 45 Replaced By...  g 1210631 pdb 1QMM1	309	6e-84
q 1127681 ep E02189 MYG_PTC   M Myoglobin  g 12295701 p J171377A	308	2e-83
q 16291702 ep E02189 MYG_PTC   M Myoglobin  g 16291702 ep P68...	300	4e-81

Gene Info

Structures

### 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 esquema de puntuación (match/mismatch) y gap penalties empleados

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

## BLAST: Basic Local Alignment Search Tool

### • RESUMEN de gapped-BLAST

- BLAST es un programa para búsqueda de secuencias similares a una sec. 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-BLASTP requiere al menos de dos palabras o hits no solapados con un score de al menos  $T$ , 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  $S_u$  (normalized ungapped 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 el número 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 (no se puede estimar  $\lambda$ ) éstas pueden obtenerse a partir de simulaciones *in silico*



**Genómica Evolutiva I**  
LCG-UNAM,  
Semestre 2018-1



Pablo Vinuesa (vinuesa@ccg.unam.mx)  
Centro de Ciencias Genómicas UNAM  
<http://www.ccg.unam.mx/~vinuesa/>

**Mini-tutorial de uso de BLAST y BLAST+ desde la línea de comandos:**

1. Generación de bases de datos (indexadas) mediante formatdb y makeblastdb
2. Interrogación de bases de datos mediante blastall -p [blastn|blastp|blastx|tblastn|tblastx] y blastn, blastp, blastx, delta-blast ...
3. Recuperación de secuencias de una base de datos usando Id's y fastacmd o blastdbcmd

Documentación de BLAST+ en NCBI  
<http://www.ncbi.nlm.nih.gov/books/NBK1762/>  
<http://www.ncbi.nlm.nih.gov/books/NBK52640/>  
<http://www.ncbi.nlm.nih.gov/books/NBK279690/>

## Uso de formatdb para generar bases de datos para NCBI-BLAST

- Por defecto, formatdb produce 3 archivos con el mismo nombre de base y con las extensiones base.nhr, base.nsq, y base.nin. Estos son archivos binarios, y en este caso se trata de bases de datos de secuencias de nucleótidos ya que la primera letra de la ext. comienza con n (p para proteína).
  - Los tres parámetros más usados para correr formatdb son:
    - i input data file (contiene una o más secuencias en formato FASTA)
    - n output file base name (if this parameter is not set, the input file name is used as base)
    - p type of file: T for protein, F for nucleic acid (True/False)
  - La opción -o produce otro conjunto de archivos requeridos para el indexado. Esta opción es esencial si se van a formatear bases de datos grandes.
  - La sintaxis básica para formatear una base de datos es:  
formatdb -i mis\_ESTs.fna -p F -n mis\_ESTs\_db -o T # para nucleótidos  
formatdb -i mis\_PROTs.faa -p T -n mis\_PROTs\_db -o T # para proteínas
- el primer comando toma un archivo FASTA de nucleótidos y crea los 3 archivos de base de datos: mis\_ESTs\_db.nhr, mis\_ESTs\_db.nsq, y mis\_ESTs\_db.nin  
y 2 archivos de indexado:  
mis\_ESTs\_db.nsd, mis\_ESTs\_db.nsi
- Las opciones (ayuda) de formatdb se llaman así: formatdb --help

## BASES DE DATOS PARA NCBI-BLAST

- BLAST usa bases de datos indexadas para acelerar la operación de búsqueda.
- Existen diversas bases de datos pre-compiladas y formateadas. La más general y extensa es la "nr" o no-redundante. Hay muchas más como: est, wgs, pat, pdb, microbial genomes o env\_nt.
- Tienen posible generar bases de datos propias usando el programa formatdb o makeblastdb. Descárgalo desde <ftp://ftp.ncbi.nih.gov/blast/> junto con los demás binarios de la suite de programas BLAST+. [en ubuntu: apt-get install ncbi-blast+ (blast2 es legacy-blast)]
- Para generar una base de datos se utilizan secuencias en formato FASTA, y con una sintaxis de identificador NCBI canónica. Por ejemplo:

```
lc|integer } estos son los formatos de las cabeceras FASTA para generar bases  
lc|string } de datos de secuencias localmente.  
gn|yourDB|ID } Puedes ver más ejemplos aquí:
```

[http://ncbi.github.io/cxx-toolkit/pages/ch\\_demo#ch\\_demo.id1\\_fetch.html\\_ref\\_fasta](http://ncbi.github.io/cxx-toolkit/pages/ch_demo#ch_demo.id1_fetch.html_ref_fasta)

Este identificador es esencial para un correcto indexado de la BD y para así poder, por ejemplo, recuperar secuencias de la BD usando listas de identificadores.

## Uso de blastall desde la línea de comandos

- Los programas blastn blastp blastx tblastn y tblastx se especifican como parámetro del comando blastall en el "BLAST antiguo" (legacy BLAST).
- Las opciones básicas y esenciales son:
  - p programa a ejecutar (blastn, blastp, blastx, tblastn, tblastx)
  - d base de datos sobre la cual buscar homólogos (creada con formatdb)
  - i input sequence file (una o más secuencias en formato FASTA)
  - o output file name # si lo prefieren pueden redirigir la salida a outputfile
- Ejemplos de sintaxis básica serían:
  - a) un análisis de blastn  
blastall -p blastn -i my\_query\_file -d my\_database -o \ my\_blast\_output.txt
  - b) un análisis de blastp  
blastall -p blastp -i my\_query\_file -d my\_database -m 8 \ -b 10 > my\_blast\_output.txt

## Otras opciones muy útiles de *blastall*

- e [valor de expectativa de corte]. El valor por defecto es 10.0. Este número tiene que especificarse en notación decimal, no exponencial, por ejemplo -e 0.001
- F filter query sequence. Por defecto esta opción implementa el filtro "DUST" que enmascara regiones de baja complejidad
- m 8 produce formato tabular de salida, muy útil para grandes conjuntos de datos
- b [número] trunca el reporte a un máximo de [número] alineamientos
- M protein substitution matrix. La matriz por defecto es BLOSUM62. Se pueden especificar: BLOSUM45, BLOSUM80, PAM30 y PAM70.
- El resto de las opciones pueden consultarse tecleando "blastall" sin más argumentos en la línea de comandos

## Recuperar secuencias de una base de datos usando *fastacmd*

- Para recuperar las secuencias especificadas en una lista de GIs a partir de una base de datos, se usa el comando *fastacmd* usando la siguiente sintaxis:

```
fastacmd -d mis_ESTs.fna -s AU108953,AU108955 -l 80
6
fastacmd -d mis_ESTs.fna -f archivo_con_GIs_a_recuperar -l 80
6
fastacmd -d mis_ESTs.fna -D
```

donde -d designa la base de datos, -s la cadena de identificador de secuencia a recuperar, y -l el no. de caracteres por línea de secuencia. Alternativamente podemos recuperar una serie de secuencias, cuyos IDs vienen especificados en un archivo (uno por línea) que se para como parámetro a la opción -f. La opción -D hace un "dump" o vertido de toda la base de datos.

## BLAST+ - el nuevo BLAST escrito en C++

Continuación (ver blast[ntp...] -h para despliegue de opciones)

BLAST	BLAST+	Descripción
blastall	blastn, blastp, ...	
-p		No existe
-i	-query	Archivo de entrada
-d	-db	Base de datos de blast
-o	-out	Nombre de archivos de salida
-m	-outfmt	Formato salida; TAB: 6 == m 8
-e	-evalue	Punto de corte para valor de Expectancia
-v	-num_descriptions	Máximo número de descripciones - hits
-b	-num_alignments	Número máximo de alineamientos
-a	-num_threads	No. de cores a usar
	-max_target_seqs	No. max. de secuencias y descripciones
-F F	-dust no   -seg no	Deshabilitar filtrado de regiones de baja complejidad; DNA:dust AA:seg

## Campos del formato tabular -m 8|9 de NCBI-BLAST

- Como ya vimos, la opción -m 8|9 de blastall especifica una salida en formato tabular, con los campos separados por tabuladores.
- Estos datos (líneas) se pueden parsear fácilmente usando Perl o comandos de UNIX como:

```
# imprime sólo hits con %ID > 95% y aln_len > 500
perl -ane '{ print "$F[0]\tF[1]" if $F[2] > 95.0 && $F[3] > 500 }' blast_m8.out
# obtén una lista no redundante de hits
cut -f2 blast_output.txt | sort -u
```
- Los campos o columnas son las siguientes: (-m 9 los imprime como comentario)
  - 0: query name
  - 1: subject name
  - 2: percent identities
  - 3: alignment length
  - 4: number of mismatched positions
  - 5: number of gap positions
  - 6: query sequence start
  - 7: query sequence end
  - 8: subject sequence start
  - 9: subject sequence end
  - 10: e-value
  - 11: bit score

## BLAST+ - el nuevo BLAST escrito en C++

REFERENCIAS CLAVES:  
 1: Boratyn GM, Canacho C, Cooper PS, Coulouris G, Fong A, Ma N, Madden TL, Matten WT, McGinnis SD, Merezuk Y, Raytelis V, Savers EN, Tao Y, Ye J, Zaretskaya I. BLAST: a more efficient report with usability improvements. *Nucleic Acids Res.* 2013 Jul;41(Web Server issue):W33-40. doi: 10.1093/nar/gkt282. Epub 2013 Apr 22. PubMed PMID: 23609542; PubMed Central PMCID: PMC3692093.

2: Canacho C, Coulouris G, Avayon V, Ma N, Papadopoulos J, Bealer K, Madden TL. BLAST+: architecture and applications. *BMC Bioinformatics.* 2009 Dec 15;10:421. doi: 10.1186/1471-2105-10-421. PubMed PMID: 20003500; PubMed Central PMCID: PMC2803857.

Conviene revisar además el **BLAST Command Line Applications User Manual**

en: <http://www.ncbi.nlm.nih.gov/books/NBK1763/>

Aquí sólo va un resumen de algunos comandos básicos, comparando blast con blast+:

BLAST	BLAST+	Descripción
formatdb	makeblastdb	
-i	-in	Archivo de entrada con secuencias
-p T/F	-dbtype prot/nucl	Mol type
-o T	-parse_seqids	Parsea e indexa seq IDs
-n	-out	Nombre de base para archivos de salida

## BLAST+ - el nuevo BLAST escrito en C++

BLAST	BLAST+	Descripción
fastacmd	blastdbcmd	
-d	-db	Base de datos de blast
-s	-entry	Cadena de búsqueda
-D 1	-entry all	DB dump en formato FASTA

Ejemplos de uso de programas de la suite de programas BLAST+

# 1) formateo de la base de datos

**makeblastdb** -in secuencias4blastdb.fna -dbtype nucl -parse\_seqids

# 2) ejecutamos una búsqueda con blastn sobre la base de datos recién formateada

**blastn** -query query\_seqs.fas -db secuencias4blastdb.fna -out 16S\_out.tab -

**# 3) formateo de la base de datos usando blastdbcmd**

**blastdbcmd** -db secuencias4blastdb.fna -entry my\_hits.list

Ya es hora de hacer unos ejercicios con datos reales...

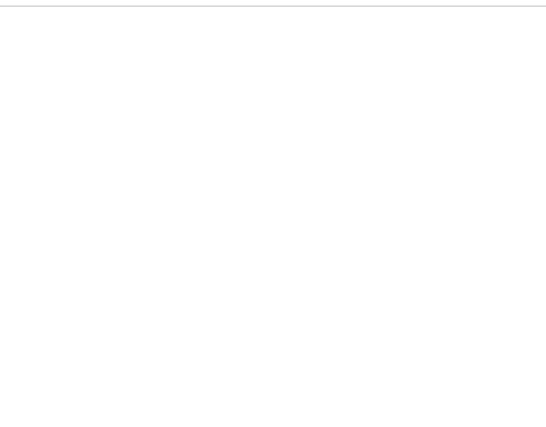
## Ejercicios: formateo de bases de datos de nt y aa con blastdb y búsquedas locales con blastall

- I. Formateo de base de datos de secuencias 16S de *Mycobacterium* spp. y búsqueda en ella de homólogos mediante blastn
  - 1) Descargar el archivo `16S_4blastN.tgz` de la página del curso
  - 2) Descomprimirlo y abrir el tarro con: `tar -xvzf 16S_4blastN.tgz`
  - 3) Construiremos la base de datos con las secuencias disponibles en el archivo `16S_seqs4_blastDB.fna`. Primero que nada averigüen:
    - 3.1 ¿cuántas secuencias tiene; cuántas especies representa?
    - 3.2 ¿qué información contienen los identificadores (el *fasta header*) ?
    - 3.3 ¿es su formato adecuado para un indexado correcto?
 Usa la línea de comandos para dar respuesta a estas preguntas
  - 4) ¿Qué línea de comando usarías para un generar una base de datos con el archivo `16S_seqs4_blastDB.fna` para que esté indexado?
  - 5) ¿Cómo clasificarías las secuencias contenidas en el archivo `16S_problema.fna` ?
  - 6) Recupera los 10 hits más próximos a cada secuencia problema de la base de datos para su posterior alineamiento; filtra aquellos hits con  $\geq 98.5\%$  de identidad

## Ejercicios: continuación

- II. Formateo de base de datos de secuencias de integrones bacterianos y descubrimiento y anotación de genes (*cassettes*) amplificados de cepas de *E. coli* recuperadas por Jazmín Madrigal del río Apatlaco, Mor. México.
  - 1) Descargar el archivo `gene_discovery_and_annotation_using_blastx.tgz` de la página
  - 2) Descomprimirlo y abrir el tarro con:
 

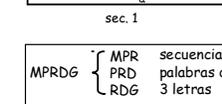
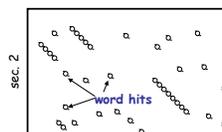
```
tar -xvzf gene_discovery_and_annotation_using_blastx.tgz
```
  - 3) Construiremos la base de datos con las secuencias disponibles en el archivo `integron_cassettes4blastdb.faa`. Primero que nada averigüen:
    - 3.1 ¿cuántas secuencias tiene; cuántas especies representa?
    - 3.2 ¿qué información contienen los identificadores (el *fasta header*) ?
    - 3.3 ¿es su formato adecuado para un indexado correcto?
 Usa la línea de comandos (shell) para dar respuesta a estas preguntas
  - 4) ¿Qué comando usarías para un generar una base de datos con el archivo `*4blastdb.faa` para que esté indexado?
  - 5) ¿Qué comandos usarías para identificar y anotar los genes que pudieran estar codificados en las secuencias contenidas en el archivo `3cass_amplicons.fna`?
  - 6) Recupera los 10 hits más próximos a cada secuencia problema de la base de datos para su posterior alineamiento.



## BLAST: Basic Local Alignment Search Tool

### • Ensemillado

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 usados como semillas de alineamientos. Así se reduce mucho el espacio a explorar.



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 ponderación.  $T$  corresponde a un umbral (Threshold) mínimo de score que han de tener las palabras encontradas.

Vecinos aceptados de RDG serían:

Palabra	Score (Blosum62)
RDG	17
KGD	14
QGD	13
RGE	13
EGD	12
...	

## 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 pares de letras con o sin gaps. Score = sumatoria de scores individuales  $p_{ab}$  - costo gaps.

BLAST no explora todo el espacio de búsqueda entre dos secuencias (es un heurístico).



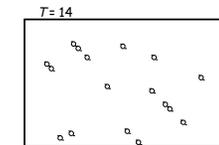
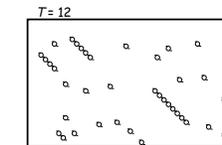
BLAST reporta todos los alns. pareados (HSPs) estadísticamente significativos encontrados en su búsqueda heurística del espacio de búsqueda. Hay que entender que en las búsquedas BLAST siempre hay que hacer un compromiso entre velocidad y sensibilidad. La velocidad se gana al no explorar toda la matriz, perdiéndose sensibilidad.

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

## BLAST: Basic Local Alignment Search Tool

### • Ensemillado

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 significativo.



El tamaño de palabra  $W$  es otro parámetro 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 de BLAST

**BLAST: Basic Local Alignment Search Tool**

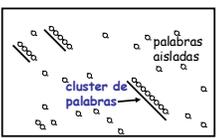
**• Ensemillado**

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 (*A*), más palabras aisladas serán ignoradas, reduciéndose consecuentemente el espacio de búsqueda, incrementándose la velocidad a costa de perder sensibilidad.

**• Detalles de implementación: BLASTN vs. BLASTP**

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

2. En **BLASTP** (y otros programas basados en aa) se 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 (*A*) entre vecinos del algoritmo two-hit es por defecto = 40 aas. Las palabras que ocurren con una frecuencia significativamente mayor que la esperada por azar (FFF) corresponden frecuentemente a **regiones de baja complejidad (rbc)** que generalmente son **enmascaradas**. El uso de "soft masking" evita el ensmellado en rbc



**BLAST: Basic Local Alignment Search Tool**

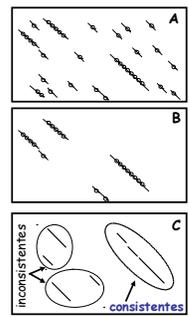
**• Evaluación**

Una vez extendidas las semillas, los **alns.** resultantes son evaluados para determinar si son **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. involucrados, 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 *A*T)**, basado en los scores de los alns. y que no considera por tanto 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**

**• Extensión**

Una vez que el espacio de búsqueda ha sido ensmellado, 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 cuánto 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

