Noname manuscript No.
(will be inserted by the editor)

Basic Local Alignment Search Tool (BLAST)

Trabajo 1 - Laboratorio de Bioinformática

Giovanni Benussi · Ismael Vicencio · Daniel Vega

Received: date / Accepted: date

Abstract La búsqueda de similitudes en las base de datos de DNA es una tarea prioritaria para la Bioinformática y la Biología molecular. Los grandes bloques de datos existentes en el DNA y el poder de computo que existe en la actualidad, no son aptos para implementar algoritmos de enfoque dinámico como el de Smith-Waterman, el cual permite resolver los alineamientos locales de forma optima en cadenas de menor tamaño, por ser poco realistas en tiempo de ejecución. Estos motivos, llevan a implementar algoritmos heurísticos como FASTA y BLAST, los cuales permiten tiempos de ejecución menores, obteniendo buenos resultados.

Keywords Alineamento · Secuencias · BLAST · Smith-Waterman

1 Introducción

La necesidad por parte de los biólogos de utilizar e interpretar grandes cantidades de datos constantemente en el área de la investigación genómica ha

G. Benussi
Universidad de Santiago de Chile. Avenida Ecuador #3659. Estación Central, Santiago de Chile.
Tel.: +56-9-74659339
E-mail: giovanni.benussi@usach.cl

I. Vicencio
Universidad de Santiago de Chile. Avenida Ecuador #3659. Estación Central, Santiago de Chile.
Tel.: +56-9-62287921
E-mail: ismael.vicencio@usach.cl

D. Vega
Universidad de Santiago de Chile. Avenida Ecuador #3659. Estación Central, Santiago de Chile.
Tel.: +56-9-79753204
E-mail: daniel.vega.a@usach.cl

dado lugar a una nueva área de estudio, llamada Bioinformática. Alguien con conocimientos en Bioinformática sirve como intermediario entre informáticos y biólogos, ya que tiene conocimientos de ambos mundos, lo cual tiene muchos beneficios, entre ellos, la correcta parametrización de programas informáticos de biología, la modelación correcta de las necesidades de biólogos al desarrollar un *software* específico, correcta interpretación de resultados obtenidos de un programa específico, optimización de algoritmos, entre otros.

2 Secuencias

2.1 Alineamiento de secuencias

1. Describa los tipos de secuencias que se pueden encontrar en Bioinformática.

   En Bioinformática se pueden encontrar distintos tipos de secuencias ([18], [16]), entre ellas se pueden mencionar:
   - **Ácido Desoxirribonucleico (ADN):** Es una molécula en forma de hélice constituída por dos cadenas paralelas de nucleótidos y soportada por unidades de grupos de fosfato y azúcar. Está compuesto por 4 tipos de bases nitrogenadas: Adenina (A), Citocina (C), Guanina (G) y Timina (T). La duplicación de este ocurre antes de la división celular, y, en esta, las cadenas de ADN se separan y son usadas como molde para la síntesis de la nueva cadena. El ADN contiene las instrucciones genéticas usadas en el desarrollo y funcionamiento de los organismos vivos (inclusive de algunos virus), además de ser el responsable de transmitir su información para que esta sea heredada.
   - **Ácido Ribunucleico (ARN):** Al igual que el ADN se compone de nucleóticos, pero difiere de este en que las bases nitrogenadas son: Adenina (A), Uracila (U), Citocina (C) y Guanina (G). Además, el ARN está compuesto de una única cadena de nucleótidos. El ARN desempeña diversas funciones, de las cuales, una de las más importantes, es dirigir las etapas intermedias de la síntesis proteica.
   - **Proteínas:** Son moléculas formadas por cadenas lineales de aminoácidos. Estas desempeñan múltiples funciones, por ejemplo, estructural, inmnológica, enzimática, protectora o defensiva, homeostática, entre otras. Una función muy importante es la manera en que participa con otras proteínas y moléculas en mantener la célula viva e interactuar con el medioambiente.

2. Defina los conceptos de alineamiento global y alineamiento local, detallando: diferencias entre ambos tipos, aplicaciones, algoritmos principales y herramientas disponibles.

   El alineamiento de secuencias nor permite cuantificar el grado de similitud entre dos secuencias determinadas mediante la alineación de

estas [19]. Existen dos tipos de alineamientos de secuencias: el alineamiento global y el alineamiento local.

(a) **Alineamiento Global:** Corresponde al alineamiento de secuencias en el cual se abarcan las secuencias completas.
(b) **Alineamiento Local:** Corresponde al alineamiento de la zona que entregue la mayor similaridad mutua.

Ambas secuencias utilizan gaps (secuencia de espacios en una de las filas del alineamiento) donde sea necesario.
Respecto a las aplicaciones de ambos, un alineamiento local es útil en el caso en el cual se sospecha que las secuencias presentan regiones similares a pesar de que sean diferenciadas globalmente. Estas regiones similares podrían representar relaciones funcionales entre las secuencias evaluadas. Por otro lado, un alineamiento global se debe utilizar cuando se sabe que dos secuencias se parecen a lo largo de toda su extensión (existen técnicas como la matriz de puntos para tener una idea de esto).
Los algoritmos principales para llevar a cabo el alineamiento global y local son:

(a) **Alineamiento Global:** Para llevar a cabo un alineamiento global, se utiliza principalmente el algoritmo de Needleman-Wunsch, propuesto en 1970 por Saul Needleman y Christian Wunsch, utiliza programación dinámica. Garantiza la obtención del mejor alineamiento.
(b) **Alineamiento Local:** Para el alineamiento local, principalmente se utiliza el algoritmo de Smith-Waterman, el cual utiliza programación dinámica para determinar regiones similares entre un par de secuencias. Fue propuesto en 1981 por Temple Smith y Michael Waterman. El alineamiento encontrado es óptimo respecto al sistema de puntajes que se utilice.

Dentro de las herramientas disponibles para realizar alineamiento global y local se encuentran:

(a) **BLAST (Basic Local Alignment Search Tool:** Es una herramienta de búsqueda de alineamientos de secuencias locales. Es capaz de comparar una secuencia dada con las secuencias que se encuentren en una base de datos, entregando las secuencias que tienen una mayor similaridad con la entregada. Utiliza un método heurístico, por lo cual no garantiza entregar una solución correcta. Pese a lo anterior, BLAST entrega la significación de sus resultados, lo cual nos permite evaluar con mayor información los resultados obtenidos. BLAST es desarrollado por los institutos nacionales de salud del gobierto de Estados Unidos y puede utilizarse gratuitamente tanto de manera online como para ser instalado de manera local.
(b) **FASTA:** Sirve para encontrar alineamientos locales o globales entre proteínas o secuencias de DNA. Utiliza un método heurístico para realizar sus trabajos. Pese a esto, si el usuario lo desea, puede indicar

que desea utilizar el algoritmo de *Smith-Waterman* para la búsqueda. Además, FASTA permite realizar búsquedas en grandes bases de datos de secuencias de DNA.

(c) **Biostring:** Paquete para el lenguaje de programación R de la librería Bioconductor que permite realizar alineamiento de múltiples secuencias. Entre otras opciones, nos permite elegir entre realizar un alineamiento global o local. Para el alineamiento global utiliza el algoritmo de Needleman-Wunsch, y para el alineamiento local, utiliza el algoritmo de Smith-Waterman, por lo cual, el consumo de recursos será mayor a medida que aumente el tamaño de las secuencias a evaluar.

3. Defina homología de secuencias: ortología y paralogía. Señale ejemplos.

Cuando dos genes de dos especies distintas derivan de un mismo gen ancestral, se dice que son homólogos [17]. Por ejemplo, al comparar el génoma de dos especies, si se encuentran genes similares, con pequeñas variaciones en sus secuencias, se puede decir que los genes vienen de un antepasado común. La Figura 1 muestra un ejemplo de Homología.

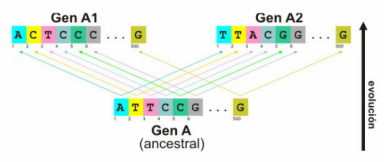

Fig. 1: Ejemplo de Homología

La ortología corresponde a las secuencias homólogas que han sido separadas por un evento de especialización, el cual ocurre cuando una especie diverge en dos separadas. En la Figura 2 se puede apreciar un ejemplo de ortología.
Por otro lado, la paralogía corresponde a una secuencia homóloga. La Figura 3 muestra un ejemplo de paralogía.

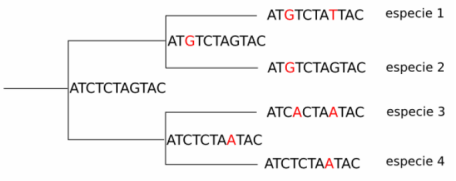

Fig. 2: Ejemplo de Ortología

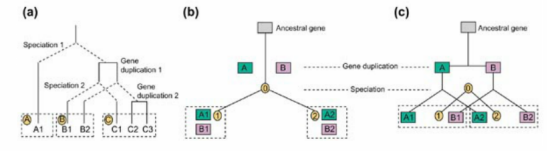

Fig. 3: Ejemplo de Paralogía

2.2 Basic Local Alignment Search Tool (BLAST)

1. Describa las principales características y aplicaciones de BLAST.

   BLAST es el acrónimo de Basic Local Alignment Search Tool, es una herramienta alternativa a los algoritmos tradicionales de búsqueda de secuencias, su desarrollo proviene del hecho que los algoritmos tradicionales precisan de una gran cantidad de tiempo para realizar comparaciones, dado por el gran tamaño de las bases de datos que son empleadas. Este algoritmo se presenta como uno de tipo heurístico, que aproxima el algoritmo de Smith-Waterman, siendo un poco menos preciso pero más de 50 veces más rápido, por lo cual su principal característica es la velocidad, realizando búsquedas en pocos minutos en la totalidad de los datos presentes en la base de datos seleccionada.
   BLAST es el algoritmo por excelencia para realizar búsquedas preliminares de similitud entre una secuencia problema y las bases de datos disponibles. Permite entregar el mejor alineamiento, así como también algunas otras subsecuencias que presentan un score similar al mejor encontrado. Es un aherramienta online, que posee diversos niveles de calibración a la hora de realizar una búsqueda, así como también niveles de puntuación para los match y mismatch, al igual que para los gap. Es una herramienta poderosa, ya que tiene una gran cantidad de alternativas de comparación, la que puede ser usada para búsquedas de nucleótidos, proteínas y nucleótidos trasladados, además de da la opción de realizar búsquedas especializadas, lo cual entrega muchas prestaciones a sus usuarios.

2. Diríjase a http://blast.ncbi.nlm.nih.gov/Blast.cgi y caracterice cada uno de los cinco programas disponibles en BasicBLAST: nucleotide blast, protein blast, blastx, tblastn, tblastx.
   (a) **Protein blast (blastp):** compara una secuencia de aminoácidos en una base de datos de secuencias de proteínas.

```
>MIMI_L4 complement(6238..7602)
MPQKTSKSKSSRTRYIEDSDDETRGRSRNRSIEKSRSRSLDKSQKKSRDK
SLTRSRSKSPEKSKSRSKSLTRSRSKSPKKCITGNRKNSKHTKKDNEYTT
```

Fig. 4: blastp - Ejemplo de secuencia de aminoácidos

   (b) **Nucleotide blast (blastn):** compara una secuencia de nucleótidos dentro de una base de datos de secuencias de nucleótidos.

```
>DNA_seq1
CGGGAGGCGGCAGCGGCTGCAGCGTTGGTAGCATCAGCATCAGCATCAGCGGCAGCGGCAGCGGCCTCGG
GCGGGGCCGGCCGGACGGACAGGCGGACAGAAGGCGCCAGGGGCGCGCGTCCCGCCCGGGCCGGCCATGG
AGGGCGCCTCCTTCGGCGCGGGCCGCGCAGGGGCCGCCCTGGACCCCGTGAGCTTTGCGCGGCGGCCCCA
```

Fig. 5: blastn - Ejemplo de secuencia de nucleótidos

   (c) **Blastx:** compara una secuencia de nucleótidos traducida en sus seis posibles marcos de lectura con una base de secuencias de proteínas.

```
>human_genomic_seq
TGGACTCTGCTTCCCAGACAGTACCCCTGACAGTGACAGAACTGCCACTCTCCCCACCTG
ACCCTGTTAGGAAGGTACAACCTATGAAGAAAAAGCCAGAATACAGGGGACATGTGAGCC
ACAGACAACACAAGTGTGCACAACACCTCTGAGCTGAGCTTTTCTTGATTCAAGGGCTAG
TGAGAACGCCCCGCCAGAGATTTACCTCTGGTCTTCTGAGGTTGAGGGCTCGTTCTCTCT
TCCTGAATGTAAAGGTCAAGATGCTGGGCCTCAGTTTCCTCTTACATACTCACCAAAAGG
CTCTCCTGATCAGAGAAGCAGGATGCTGCACTTGTCCTCCTGTCGATGCTCTTGGCTATG
ACAAAATCTGAGCTTACCTTCTCTTGCCCACCTCTAAACCCCATAAGGGCTTCGTTCTGT
GTCTCTTGAGAATGTCCCTATCTCCAACTCTGTCATACGGGGGAGAGCGAGTGGGAAGGA
TCCAGGGCAGGGCTCAGACCCCGGCGCATGGACCTAGTCGGGGGCGCTGGCTCAGCCCCG
CCCCGCGCGCCCCCGTCGCAGCCGACGCGCGCTCCCGGGAGGCGGCGGCAGAGGCAGCAT
CCACAGCATCAGCAGCCTCAGCTTCATCCCCGGGCGGTCTCCGGCGGGGAAGGCCGGTGG
GACAAACGGACAGAAGGCAAAGTGCCCGCAATGGAGGGAGCATCCTTTGGCGCGGGCCGT
GCGGGAGCTGCCTTTGATCCCGTGAGCTTTGCGCGGCGGCCCCAGACCCTGTTGCGGGTC
GTGTCCTGG
```

Fig. 6: blastx - Ejemplo de secuencia de nucleótidos traducida

(d) **Tblastn:** compara una secuencia de aminoácidos con toda la base de datos de nucleótidos traducida en sus seis posibles marcos de lectura.

```
>Q9BDJ6|GHRL_BOVIN Appetite-reg.hormone precursor
MPAPWTICSLLLLSVLCMDLAMAGSSFLSPEHQKLQRKEAKKPSGRLKPRTLEGQFDPEV
GSQAEGAEDELEIRFNAPFNIGIKLAGAQSLQHGQTLGKFLQDILWEEAEETLANE
```

Fig. 7: tblastn - Ejemplo de secuencia de aminoácidos

(e) **Tblastx:** compara las seis traducciones en sus marcos de lectura de la secuencia de nucleótidos respecto a las seis traducciones en sus marcos de lectura de toda la base de datos de nucleótidos existente.

```
>gi|50539273|gb|CO636075.1 Gregarina niphandrodes
GCCATTACGGCCGGGGGAAAGAAGTACTACCAACACAATGGTGGCTGATGACAAGATAATCACCACGAAG
GTTGTTTCTAAGAAGGTGGTTTCCACCAAGGTGGTCACCGGCGGCGACAAGAAGAAGAAGCATGTTGTTG
CCGAATAATTGATCTACATTATTAACTTGTTCATCAACATCCTCAACGACCTCATCATCATCTCCTACAC
GATGATCATCGTACACAACTTCTTACTCGACAACTTCTCCTCAATCTTCTATGATGCCACCAACCTCTGC
TGCCGCACCACCACTAATACAATTGTTGCTCGTCAAATCCTCAACGCCGTCCGTCTTATCATCCTCGCCG
AACTGTTGACCCACGCTACCACCAACGGGGCCACCGCCGCCACCCACTACTACAGACCCGCCTACCAACG
TTGTACATACATATTCTACATCATCAACGACTGATGCCACCGAGCTTCTTTCCACGCCTGATCTACTTGC
CCACCCAACAACCCCCCTAAAACAAAATCTGTGTCCCCCGTCTCGGCCCGCGTATCCAATATTTCTAGAG
CCGCCCCCACCGGTGCGGATCCCCCTCTTGTGCCCCCCTTCACGGGGGTTTATTCCCGCCCGTGGGTGTA
CCACAGTGCCCCACACCTGGTCCCCTGGTGCACAAACGTGCTTACCCCCTCTACAC
```

Fig. 8: tblastx - Ejemplo de secuencia de nucleótidos traducida

A continuación se presenta una tabla resumen

Search page	Query & database combination	Alignment type	Programs & functions (default program in bold)
nucleotide blast	nucleotide vs nucleotide	nucleotide vs nucleotide	**megablast**: for sequence identification, intra-species comparison discontiguous megablast: for cross-species comparison, searching with coding sequences blastn: for searching with shorter queries, cross-species comparison
protein blast	Protein vs protein	protein vs protein	**blastp**: general sequence identification and similarity searches DELTA-BLAST [2] : protein similarity search with higher sensitivity than blastp PSI-BLAST: iterative search for position-specific score matrix (PSSM) construction or identification of distant relatives for a protein family PHI-BLAST: protein alignment with input pattern as anchor/constraint
blastx	nucleotide (translated) vs protein	protein vs protein	blastx: for identifying potential protein products encoded by a nucleotide query
tblastn	protein vs nucleotide (translated)	protein vs protein	tblastn: for identifying database sequences encoding proteins similar to the query
tblastx	nucleotide (translated) vs nucleotide (translated)	protein vs protein	tblastx: for identifying nucleotide sequences similar to the query based on their coding potential

3. Describa las aplicaciones disponibles en specialized BLAST.

BLAST suministra tipos de búsquedas especializadas, así como de bases de datos, tales como:

(a) SmartBLAST: Proporciona resultados rápidos de búsqueda de proteínas, en un entorno gráfico. Realiza una búsqueda de una secuencia de proteína en las bases de datos de proteínas.

(b) Primer-BLAST: Construye primers específicos para la reacción en cadena de la polimerasa (PCR), para ello utiliza Primer3 y BLAST.

(c) MOLE-BLAST: es una herramienta que ayuda a los taxónomistas a encontrar las bases de datos vecinas más cercanas de acuerdo a la secuencia de consulta presentada. Para ello calcula una alineación de secuencias múltiples (MSA) entre las secuencias de mejor ranking en la base de datos de aciertos de BLAST, y genera un árbol filogenético. Si las secuencias de entrada provienen de diferentes genes o loci, MOLE-BLAST puede agruparlas y calcular una MSA y un árbol filogenético para cada locación por separado.

(d) Encuentra dominios conservados (es un recurso de anotación de proteínas que consiste en una colección de múltiples modelos de alineamiento de secuencias bien estipulado para los dominios ancestrales y proteínas de longitud completa) dentro de la secuencia de búsqueda (cds).

(e) Encuentra secuencias con arquitectura de dominios conservados similares (cdart)

(f) Busca secuencias que tengan perfiles de expresión de genes (GEO)

(g) Busca inmunoglobulinas y secuencias de receptores de células T (IgBLAST)

(h) VecScreen: Es un sistema que encuentra segmentos de una secuencia de ácido nucleico que pueden ser de origen vectorial. Ayuda a los investigadores a identificar y eliminar los segmentos de origen del vector antes de analizar o enviar secuencias.

(i) Alinear dos o más secuencias usando BLAST (bl2seq)

(j) Búsqueda objetivo de proteínas o nucleótidos en PubChem BioAssay

(k) Búsqueda de SRA por experimento

(l) Restricciones basadas en herramientas de alineamiento de proteínas múltiple

(m) Needleman-Wunsch: Herramienta de alineamiento global de secuencias

(n) Búsqueda RefSeqGene

(o) Search trace archives

(p) Search bacterial and fungal rRNA sequences with Targeted Loci BLAST

4. Ingrese a nucleotide blast, familiarícese con los parámetros en pantalla y describa los parámetros encontrados en la sección algorithm parameters.

(a) Parámetros generales (General Parameters)
   i. Max target sequences: Corresponde al número máximo de secuencias alineadas que se van a desplegar por pantalla.
   ii. Short queries: Permite ajustar automáticamente algunos parámetros propios de BLAST para obtener mejores resultados en consultas de secuencias cortas.
   iii. Expect threshold: Número esperado de matches en un modelo aleatorio.
   iv. Word size: Longitud de la semilla que inicia una alineación.
   v. Max matches in a query range: Limita el número de match en una consulta de una secuencia. Es una opción útil si existen muchas coincidencias con una parte de la consulta que pueden impedir a BLAST encontrar una secuencia adecuada.

(b) Parámetros de puntaje (Scoring Parameters)
   i. Match/Mismatch Scores: Base de puntuación y penalidad para los match (1,2 y 5) y mismatch (-1,-2,-3,-4,-5) encontrados.
   ii. Gap Costs: Penalización por crear o ampliar una brecha en una alineación. Existen 8 tipos, sin embargo, los costos lineales están disponibles sólo con megablast y están determinados por las puntuaciones de match y mismatch encontradas.

(c) Filtros y máscaras (Filters and Masking)
   i. Filter: Existen dos opciones. (1) Realizar un cubrimiento de aquellos trozos de secuencias que pueden causar resultados falsos o engañosos. (2) Enmascara elementos de repetición de las secuencias especificadas que pueden conducir a resultados falsos o engañosos.
   ii. Mask: También cuenta con dos opcíones. (1) Realiza una máscara de la consulta mientras se generan las semillas para escanear la base de datos. (2) Procesa todas las letras minúsculas en la entrada FASTA.

3 Actividad Práctica

3.1 Basic Local Alignment Search Tools

La actividad corresponde al alineamiento local de secuencias utilizando la herramienta BLAST. En una primera etapa, se procede a caracterizar

las secuencias utilizando el software R. Las etapas posteriores consisten en comparar estas secuencias en diversos escenarios para conocer su comportamiento.

3.1.1 Caracterización de las secuencias

Las herramientas presentes en la librería Biostring permiten obtener información sobre la secuencia de estudio.
La secuencia de estudio presente en el archivo "secuencias.fasta", tiene una longitud de 699.860 nucleótidos, con un promedio de 174815 caracteres de cada letra del alfabeto del ADN. La subsecuencia consecutiva mas larga del alfabeto corresponde al nucleótido A con una tamaño de 33 elementos consecutivos. Respecto a los dinucleótidos, el que presenta una mayor frecuencia corresponde al dinucleótido "AA", con 58374 apariciones. Toda la información se detalla en las siguientes tablas:

Table 1: Frecuencia del Alfabeto

Alfabeto	Frecuencia
A	194672
C	144833
G	148302
T	211452

Table 2: Frecuencia de nucleotidos

Di-nucleotido	Frecuencia
AA	58374
AC	35157
AG	52128
AT	49010
CA	48784
CC	36909
CG	5575
CT	53565
GA	44269
GC	26569
GG	37830
GT	39633
TA	43242
TC	46197
TG	52769
TT	692641

Table 3: Secuencia consecutiva más larga del alfabeto

Alfabeto	Secuencia mas larga
A	33
C	14
G	17
T	28

3.1.2 Comparación inicial con BLAST

La comparación inicial consiste en evaluar la secuencia contenida en el archivo secuencia.fasta consigo misma utilizando la herramienta Blast. Al realizar la comparación, se detecta un alineamiento significativo con la secuencia presente en *Rattus norvegicus strain partialchromosome Y*, en un grado de 100%. El alineamiento es total y la herramienta BLAST indica esto al encontrar una identidad total del 100% con un total de 688.532 puntos. Los gap, como se espera en un secuencias idénticas, no existen en este alineamiento.

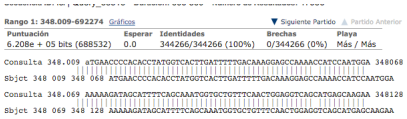

Fig. 9: Alineamiento inicial con la herramienta BLAST

3.1.3 Comparación con Genoma de Rata

Esta etapa consiste en comparar la secuencia del archivo secuenciast.fasta con el presente en el genoma del cromosoma Y del *rattus norvegicus* mediante el uso de la herramienta BLAST.
El tamaño de la secuencia del *rattus norvegicus* tiene un tamaño de 3.310.458 de nucleotidos frente a los 699.860 del archivo de estudio, es decir, un 473% mas grande. La ejecución nos presenta un gen de estudio, el cual puede ser encontrado completamente en el gen del *rattus norvegicus*. La secuencia de estudio es 100% parte del cromosoma Y, debido a que no presenta ninguna gap o separación en su alineamiento.

Fig. 10: Alineamiento de secuencia de estudio con Cromosoma Y *rattus norvegicus*

3.1.4 Comparación con organismo

Esta etapa consiste en comparar la secuencia de estudio con el organismo *Rattus norvegicus (taxid:10116)*. En la figura 11 se puede apreciar la distribución de los hits de BLAST

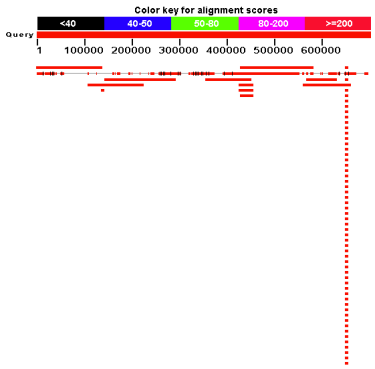

Fig. 11: Hits de la herramienta Blast

El resultado nos lleva a un alineamiento local con un score de 313110, en una secuencia de largo 156555 la cual es continua, es decir, no posee ningún gap.

Fig. 12: Alineamiento de secuencia de estudio *rattus norvegicus* (taxid:10116)

3.1.5 Análisis de Resultados

El resultado de la alineaciones entre la misma secuencia no sorprende con su igualdad de 100%, pero nos permite comprobar de forma empírica la capacidad de la herramienta BLAST y su nivel de confiabilidad. En base a esta etapa, se procede a comparar esta secuencia contra el genoma del cromosoma Y presente en un archivo fasta. Los resultados obtenidos depende de los valores de match y mismatch , 2 y 3 respectivamente. Los valores de gap existence y extension se establecieron en 5 y 2 respectivamente.
Con los valores establecidos, los resultados de la segunda ejecución nos lleva a encontrar un alineamiento total de las secuencias, con un 100% de identidad entre ellas. Con este resultado, se puede concluir que el gen de estudio es parte del cromosoma Y del *rattus norvegicus*.
En la etapa final, al analizar el alineamiento entre la especie rattus norvegicus

(taxid:10116) y nuestra secuencia de estudio, esta sólo se encuentra de forma parcial, en un máximo del 45% apróximadamente. Extrapolando esta información, podemos afirmar que la secuencia de estudio del cromosoma Y del supuesto individuo "rattus norvegicus" presenta algún tipo de mutación. Esta afirmación puede definir a nuestra secuencia de estudio como un elemento de la evolución natural o algún tipo de enfermedad.

3.2 Alineamiento local en R

La actividad correspondiente al alineamiento local de secuencias se realiza en 3 etapas, en las cuales se utilizarán herramientas distintas con el objetivo de comparar sus resultados y su rendimiento respectivamente.
Las secuencias a analizar son las siguientes:
seq1 ¡- "GAATTCCTACTACGAAGAATTCCTACTACGAAACTACGAAAATTCCTACTACGA"
seq2 ¡- "GAATTCCTACTACGGAATTCCCCTCCCATAATTCCTACTACGA"
Para el desarrollo del alineamiento se utilizarán los siguientes parámetros para todas las herramientas a utilizar:

Table 4: Frecuencia del Alfabeto

Parámetro	Valor
Match	1
MisMatch	-2
GapOpening	0
GapExtension	4

3.2.1 Alineamiento local con Blast

La primera etapa corresponde al alineamiento local utilizando la herramienta BLAST. Esta herramienta posee 3 algoritmos distintos, cada cual esta optimizado para situaciones particulares. Estos algoritmos son megablast , discontiguous blast y blastn.
Al utilizar los algoritmos megablast y discontiguous blast el programa no entrega ningún resultado, solo la ventana que se entrega a continuación:

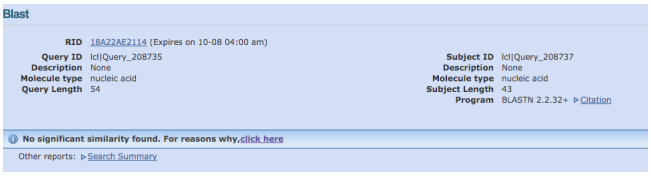

Fig. 13: Resultado al ejecutar la herramienta BLAST con megablast y discontiguous blast

Lo anterior se debe a que la secuencia es muy pequeña, por lo cual, el algoritmo al estar optimizado para secuencias de gran tamaño, no es capaz de analizar la entrada dada.
La ejecución con el algoritmo de blastn, nos entrega 3 resultados posibles.

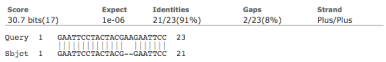

Fig. 14: Alineación número 1 al ejecutar la Herramienta BLAST

En este resultado, se obtiene un score de 17 puntos, considerando que esta configuración de alineamiento posee 21 match y 2 gap. Considerando estos elementos, se puede determinar un nivel de similitud del 91% para este alineamiento.

Fig. 15: Alineación número 2 al ejecutar la Herramienta BLAST

El segundo alineamiento obtenido con BLAST, nos entrega un score de 16 punto, pero con un largo de secuencia mucho mayor si se compara con el primer resultado. Esta configuración posee 5 gap , 35 match y 2 mismatch en la alineación.

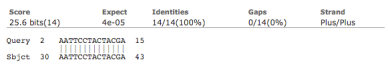

Fig. 16: Alineación número 3 al ejecutar la Herramienta BLAST

La última alineación obtenida con BLAST, posee un score de 14 punto, pero además posee un largo mas reducido que las anteriores configuraciones y esta construida solo por Match.

3.2.2 Alineamiento local con Biostring

La segunda etapa corresponde a utilizar el algoritmo de Smith-Waterman presente en la librería Biostring de Bioconductor. Para su implementación, se requiere del uso de dos funciones presentes en la librería Biostring, las cuales son nucleotideSubstitutionMatrix y pairwiseAlignment.
La función pairwiseAlignment permite aplicar alineamientos globales (Needleman-Wunsch) y locales (Smith-Waterman). Por defecto la función esta diseñada para alineamientos de tipo global, por lo que se requiere dar un argumento a esta función para utilizar el alineamiento local que se desea implementar. El modelo de la función es el siguiente:

```
pairwiseAlignment(pattern, subject,
                  patternQuality = PhredQuality(22L), subjectQuality = PhredQuality(22L),
                  type = "global", substitutionMatrix = NULL, fuzzyMatrix = NULL,
                  gapOpening = -10, gapExtension = -4, scoreOnly = FALSE)
```

Fig. 17: Función pairwiseAlignment de Biostring

Los argumentos son: pattern : Corresponde a un vector de tamaño n, con $n \geq 0$. En el caso de estudio corresponde a la primera secuencia a alinear.
subject: Corresponde a un vector de tamaño m , con m ¿ 0. En el caso de estudio corresponde a la segunda secuencia a alinear.
PatternQuality y subjectQuality : Corresponden a la calidad deñ puntaje obtenido para las secuencias a alinear. En nuestro caso este valor no considerado.
Type: Corresponde al tipo de alineamiento que se desea implementar. Por defecto el alineamiento es global. En el caso de estudio se debe utilziar type=”local” para implementar el algoritmo de Smith-Waterman
substitutionMatrix: Corresponde a la matriz de sustitución para el alineamiento cuando las secuencias no tienen una calidad esperada. En el caso de estudio, se implementa mediante la función nucleotideSubstitutionMatrix con valores match y mismatch de 1 y -2 respectivamente.
FuzzyMatrix: Matriz de match para secuencias con calidad esperada. Esta formada por valores entre 0 y 1 donde el valor 0 corresponde a un mismatch y el valor 1 aun match- Los valores que están entre 0 y 1 representan la ambigüedad que existe entre match y mismatch. No es utilizada en el caso de estudio. GapOpening: Corresponde al costo de iniciar un gap en el alinemiento. En el caso de estudio presenta un valor 0.
GapExtension: Corresponde al costo a lo largo de un gap. En el caso de estudio se utiliza un valor de -2 , debido a que es el mínimo valor que nos permite utilizar la herramienta de BLAST.
ScoreOnly: Corresponde a la salida de la función de alineamiento. En el caso verdadero muestra solo el puntaje obtenido por la alineación obtenida. En caso contrario, muestra el puntaje obtenido y el diagrama de la alineación obtenida.
La función nucleotideSubstitutionMatrix construye una matriz para todos los ácidos nucleótidos basados en los parámetros de match y mismatch, ya sea

para ADN o ARN. Su modelo es el siguiente:

```
nucleotideSubstitutionMatrix(match = 1, mismatch = 0, baseOnly = FALSE, type = "DNA")
```

Fig. 18: Función nucleotideSubstitutionMatrix de Biostring

Sus argumentos son:
Match: El puntaje para un match entre nucleótidos. En el caso de estudio se utiliza con valor 1.
Mismatch: El puntaje de un mismatch entre nucleótidos. En el caso de estudio se utiliza con valor -2.
baseOnly: Corresponde a los elementos que forman la matriz de sustitución. Es caso verdadero se utilizan las bases del alfabeto, como puede ser "A", "C", "G", "T".
type: Corresponde al tipo de elemento a utilizar. Puede ser DNA o RNA.
Finalmente, la implementación del algoritmo de Smith-Waterman utilizando las funciones presentes en Biostring se resume en las siguientes dos instrucciones.

```
mat<-nucleotideSubstitutionMatrix(match = 1, mismatch = -2, baseOnly = TRUE)
pairwiseAlignment(pattern = seq2, subject = seq1, type="local",substitutionMatrix=mat, gapOpening=0, gapExtension=-2)
```

Fig. 19: Función de Smith-Waterman de Biostring

La matriz de sustitución contiene la siguiente información:

```
> mat
  A  C  G  T
A  1 -2 -2 -2
C -2  1 -2 -2
G -2 -2  1 -2
T -2 -2 -2  1
```

Fig. 20: Matriz de sustitución del algoritmo de Smith-Waterman

Al aplicar el algoritmo sobre las secuencias de estudio se obtienen los siguientes resultados:

```
Local PairwiseAlignmentsSingleSubject (1 of 1)
pattern: [1] GAATTCCTACTACG--GAATTCC
subject: [1] GAATTCCTACTACGAAGAATTCC
score: 17
```

Fig. 21: Resultado del algoritmo de Smith-Waterman con Biostring

3.2.3 Alineamiento local con Algoritmo implementado

La tercera etapa corresponde a la implementación de un algoritmo escrito en R que permita alinear secuencias mediante el método de Smith-Waterman sin utilizar las funciones presentes en la librería Biostring. Los algoritmos correspondes a los alumnos Ismael Vicencio, Daniel Vega y Giovanni Benussi respectivamente.

1. Algoritmo 1

Al aplicar el algoritmo sobre las secuencias de estudio se obtienen, además de una alineación y su correspondiente score, la matriz de valor y sentido que se requieren para construir la solución final. Las matrices del caso de estudio son de un gran tamaño, por lo que a modo de explicar el funcionamiento del algoritmo, se dará de ejemplo las matrices para el análisis de las siguientes secuencias:
seq1 ¡- "CGTGAATTCAT"
seq2 ¡- "GACTTAC"
La primera matriz corresponde al valor que se obtiene al comparar cada nucleótido. Los valores se obtienen de acuerdo a la siguiente fórmula:

$$M_{i,j} = Maximum\ [M_{i-1,j-1} + S_{i,j},\ M_{i,j-1} + W,\ M_{i-1,j} + W, 0]$$

Fig. 22: Ecuación de Smith-Waterman

Donde:
M = valor de la matriz para la posición de estudio i , j.
S = valor de match o mismatch.
W = valor de un gap.
La matriz que genera el algoritmo es la siguiente:

	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12
1	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	1	0	1	0	0	0	0	0	0	0
3	0	0	0	0	0	2	1	0	0	0	1	0
4	0	1	0	0	0	0	0	0	0	1	0	0
5	0	0	0	1	0	0	0	1	1	0	0	1
6	0	0	0	1	0	0	0	1	2	0	0	1
7	0	0	0	0	0	1	1	0	0	0	1	0
8	0	1	0	0	0	0	0	0	0	1	0	0

Fig. 23: Matriz de valores del algoritmo 1

La segunda matriz que genera el algoritmo corresponde a la dirección de precedencia del valor en cada una de las posiciones de la matriz inicial. Esta matriz esta formada por los valores 0, 1, 2, y 3. Cada valor corresponde a una dirección, a excepción del valor 0 que corresponde a un termino en la secuencia. Los valores 1, 2 y 3 corresponden a la diagonal, up y left respectivamente. Como ejemplo se muestra una matriz de prueba.

	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12
1	0	0	0	0	0	0	0	0	0	0	0	0
2	0	1	1	1	1	1	1	1	1	1	1	1
3	0	1	1	1	1	1	1	1	1	1	1	1
4	0	1	1	1	1	2	1	1	1	1	1	1
5	0	1	1	1	1	1	1	1	1	1	1	1
6	0	1	1	1	1	1	1	1	1	3	1	1
7	0	1	1	1	1	1	1	1	2	1	1	1
8	0	1	1	1	1	1	1	1	1	1	1	1

Fig. 24: Matriz de dirección del algoritmo 1

Los resultados al implementar el algoritmo sobre las secuencias de estudio es el siguiente:

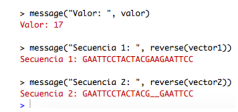

Fig. 25: Resultado alineamiento local del algoritmo 1

2. Algoritmo 2

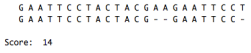

Fig. 26: Resultado alineamiento local del algoritmo 2

3. Algoritmo 3

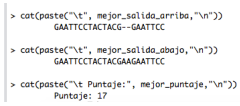

Fig. 27: Resultado alineamiento local del algoritmo 3

3.2.4 Analisis de Resultados

Las 3 pruebas ejecutadas en está etapa obtuvieron el mismo resultado de forma global, a excepción de la ejecución del algoritmo 2, el cual obtiene un resultado más bajo si se compara con la versión del algoritmo presente en la librería Biostring. Al conocer esta información, se puede determinar que la implementación del algoritmo de Smith-Waterman sin utilizar la librería Biostring, esta bien construida en lo referente a los resultados finales solo con la diferencia en los tiempos de ejecución de ambas. Este tiempo no es perceptible en secuencias de tamaños pequeños (100 nucleotidos) pero a medida que su tamaño crece, la diferencia entre estas dos implementaciones es considerable. En respecto al tiempo de ejecución, es difícil determinar el tiempo que utiliza la herramienta BLAST, ya que al ser una herramienta web, se deben considerar otros factores que son externos a la herramienta.
Si se analiza la situación que ocurre con el algoritmo número 2, este resultado es igual a la tercera alineación obtenida con BLAST sólo a nivel de score. Esto se debe a los criterios de recorrido que posee el algoritmo 2. Esta solución, aunque sea de tamaño menor, tiene una gran importancia al considerar que obtiene una identidad del 100% con la secuencia de estudio, es decir, no presenta ningún gap en ella.
El tamaño de las alineaciones obtenidas, depende de los valores con los cuales

trabajan estos algoritmos, y que permiten distribuir estas alineaciones en bloques más grandes. Esto principalmente se puede analizar en la situación en que el valor del match tiende a 5 mientras que el valor de gap tiende a -4, los valores de score de las alineaciones y largo de la secuencia son de 155 y 54 respectivamente.

References

1. Bioinformatics at COMAV, *Búsqueda de secuencias en bases de datos*, https://bioinf.comav.upv.es/courses/intro_bioinf/blast.html.
2. Open wet ware, *Wikiomics:BLAST tutorial*, http://openwetware.org/wiki/Wikiomics:BLAST_tutorial#blastn.
3. http://www.sp.uconn.edu/~mcb232vc/blast.html.
4. Computer Applications in Molecular and Cell Biology, *Basic Local Alignment Search Tool (BLAST)*, http://www.sp.uconn.edu/~mcb232vc/blast.html
5. BLAST, *BLAST Assembled Genomes*, http://blast.ncbi.nlm.nih.gov/Blast.cgi
6. J. Ye, G. Coulouris, I. Zaretskaya, I. Cutcutache, S. Rozen and T. Madden, *Primer-BLAST: A tool to design target-specific primers for polymerase chain reaction*, BMC Bioinformatics.
7. Global Computational Resources for Bioinformatics Research, *Specialized BLAST and BLAST-related algorithms*, http://www.bioinformatics.utep.edu/BIMER/tools/BLAST.html.
8. IgBLAST, *IgBLAST*, http://www.ncbi.nlm.nih.gov/igblast/intro.html.
9. VecScreen, *About VecScreen*, http://www.ncbi.nlm.nih.gov/tools/vecscreen/about/.
10. Virtual Labs at Amrita Vishwa Vidyapeetham, *Smith-Waterman Algorithm - Local Alignment of Sequences*,http://vlab.amrita.edu/?sub=3&brch=274&sim=1433&cnt=1.
11. R Documentation for Package Biostrings version 2.10.22, *Scoring matrices*, http://svitsrv25.epfl.ch/R-doc/library/Biostrings/html/substitution_matrices.html
12. R Documentation for Package Biostrings version 2.10.22, *Optimal Pairwise Alignment*, http://svitsrv25.epfl.ch/R-doc/library/Biostrings/html/pairwiseAlignment.html.
13. GO2MSIG ,*Rattus norvegicus (Rat - taxon 10116) GO derived MSigDB format gene sets for use with GSEA*, http://www.go2msig.org/cgi-bin/prebuilt.cgi?taxid=10116.
14. The Universal Protein Resource ,*Taxonomy - Rattus norvegicus*,http://www.uniprot.org/taxonomy/10116.
15. Rob Knight, *Comparison of methods for estimating the nucleotide substitution matrix*,Department of Applied Mathematics University of Colorado, http://www.biomedcentral.com/1471-2105/9/511.
16. Jacques Cohen, *MBioinformatics - An Introduction for Computer Scientists*, Brandeis University.
17. Tomás Albán, *Homología, Ortología, Paralogía*, Prezi.
18. J. Medina, F. Garzón, P. Tafurth y J. Barbosa, *Recopilación Bioinformática*, Universidad Distrital Francisco José de Caldas.
19. Environmental and Computational Chemistry Group, *alineamiento de secuencias de aminoacidos*, Faculty of Pharmaceutical Sciences.