Ensamblaje *de novo* y anotación génica del genoma de *Leishmania major* mediante secuenciación masiva

Sandra González de la Fuente
Área 31. Estadística y Bioinformática.

Guillem Ylla Bou
David Merino Arranz.

5/6/2018

B) GNU Free Documentation License (GNU FDL)

C) Copyright

FICHA DEL TRABAJO FINAL

Título del trabajo:	*Ensamblaje de novo y anotación génica del genoma de Leishmania major mediante secuenciación masiva*
Nombre del autor:	*Sandra González de la Fuene*
Nombre del consultor/a:	*Guillem Ylla Bou*
Nombre del PRA:	*David Merino Arranz*
Fecha de entrega (mm/aaaa):	*06/2018*
Titulación:	*Máster Bioinformática y Bioestadística*
Área del Trabajo Final:	*Estadística y Bioinformática.*
Idioma del trabajo:	*Castellano*
Palabras clave	*Leishmania, NGS, ensamblaje de novo.*

Resumen del Trabajo (máximo 250 palabras): *Con la finalidad, contexto de aplicación, metodología, resultados i conclusiones del trabajo.*

La leishmaniasis es una enfermedad parasitaria producida por protozoos del género *Leishmania*. *Leishmania major* es la especie prototípica asociada con la leishmaniasis cutánea en el Viejo Mundo siendo la forma más frecuente de leishmaniasis. La obtención de una secuencia genómica fiable es fundamental para estudios moleculares conducentes al desarrollo de estrategias de control de la leishmaniasis. A pesar de que el genoma de referencia de *L. major* está ensamblado en las bases de datos en 36 cromosomas, datos publicados recientemente evidenciaron la existencia de errores, debido a colapsos provocados por la existencia de regiones repetitivas.

El objetivo de este trabajo fue generar un ensamblaje mejorado del genoma de *L. major*, secuenciando el genoma mediante lecturas de PacBio e Illumina y empleando diversas estrategias de ensamblaje: ensamblajes no-híbridos e híbridos (combinando ambas lecturas). Los ensamblajes no-híbridos con lecturas de PacBio generaron los mejores resultados. Sin embargo, las lecturas de Illumina han sido esenciales para extender extremos de cromosomas y corregir regiones homopoliméricas, donde se ha encontrado que PacBio tiene limitaciones.

La comparativa entre el nuevo ensamblaje y el genoma de referencia, ha evidenciado la existencia de 167 regiones que abarcan 175 genes afectados en la referencia.
Como resultado, se ha conseguido ensamblar la secuencia completa y sin fisuras de los 36 cromosomas del genoma de *L. major*. Además, se ha establecido el procedimiento bioinformático para, partiendo de datos de secuenciación por dos metodologías de NGS, sea posible completar el ensamblaje de cualquier organismo con similar o inferior complejidad genómica que *Leishmania.*

Abstract (in English, 250 words or less):

Leishmaniasis is a parasitic disease caused by protozoa of the genus *Leishmania*. *Leishmania major* is the most common form of leishmaniasis and is associated with cutaneous leishmaniasis in the Old World and it is the most frequent form of leishmaniasis. Obtaining a reliable genomic sequence is essential for molecular studies leading to the development of leishmaniasis control strategies. Although the genome of *L. major* is found assembled in 36 chromosomes in the databases, recently published data showed the existence of errors, due to collapses caused by the existence of repetitive genomic regions.

The main purpose of this work was to generate an improved genome assembly of *L. major*, sequencing the genome with PacBio and Illumina technology and using diverse assembly strategies: non-hybrid and hybrid assemblies (combining both types of reads). The non-hybrid assemblies with PacBio reads showed the best results. However, Illumina reads have been essential for extend ends of chromosomes and correct homopolymer regions, where PacBio has been found to have limitations.

The comparison between the new assembly against the reference genome, has shown the existence of 175 genes included in 167 regions affected in the reference.

As a result, the complete and seamless sequence of the 36 chromosomes of the *L. major* genome has been assembled. In addition, the bioinformatic procedure has been established so that, from sequencing data by two NGS methodologies, it is possible to complete the assembly of any organism with similar or inferior genomic complexity than *Leishmania*.

Índice

Índice de figuras

Índice de Tablas

1. Introducción

1.1 Contexto y justificación del Trabajo

La leishmaniasis es una enfermedad parasitaria producida por protozoos del género *Leishmania*. Existe un amplio espectro clínico de la leishmaniasis que abarca infecciones subclínicas (asintomáticas), lesiones cutáneas autocontroladas y formas diseminadas (leishmaniasis cutánea difusa, mucosa o visceral). *Leishmania major* es la especie prototípica asociada con la leishmaniasis cutánea en el Viejo Mundo y esta es la forma más frecuente de leishmaniasis. Esta especie produce lesiones cutáneas, sobre todo ulcerosas, que dejan cicatrices de por vida y son causa de discapacidad grave. Las afecciones de la mucosa (también conocida como leishmaniasis mucocutánea) conduce a la destrucción parcial o completa de las membranas mucosas de la nariz, la boca y la garganta y son características de la infección por *Leishmania braziliensis*. Por otro lado, *Leishmania donovani* y *Leishmania infantum* son los agentes causantes de la leishmaniasis visceral que se caracteriza por episodios irregulares de fiebre, pérdida de peso, hepatoesplenomegalia y anemia. En más del 95% de los casos es mortal si no se trata.

Las estrategias de control para la leishmaniasis pueden mejorarse mediante investigaciones a escala genómica, y para es fundamental disponer de una secuencia genómica fiable. Estos microorganismos eucariotas y diploides poseen una arquitectura cromosómica atípica con un gran número de secuencias del ADN repetitivas dispersas a lo largo de sus genomas [1][2] y una regulación de la expresión génica que se produce casi exclusivamente a nivel postranscripcional. Los genes que codifican proteínas carecen de intrones y se encuentran integrando grupos de genes (DGC, *directional gene clusters*) formados por decenas o centenas de genes, con funciones frecuentemente no relacionadas entre sí, y situados en la misma cadena de ADN [3]. Las unidades DGC están separadas por unas secuencias cortas llamadas regiones de cambio de hebra, y son los sitios en los que empiezan o acaban estas largas agrupaciones génicas [4]. La transcripción en Leishmania es policistrónica, y los genes presentes en la DGC se transcriben en grandes unidades transcripcionales policistrónicas. Asimismo, Leishmania presenta aneuploidía y con frecuencia posee amplificaciones extracromosómicas llevadas a cabo mediante recombinación homóloga en secuencias repetitivas. Además, posee un número de copias de genes variable [5]. La publicación en 2005 [6] de la secuencia del genoma de *L. major* (cepa Friedlin) y dos años más tarde los genomas para la especie *L. braziliensis* y *L. infantum*, utilizando técnicas clásicas de secuenciación, fue valiosa para, entre otras cosas, poner de manifiesto estos aspectos moleculares tan peculiares. De los mencionados, el genoma de *L. major* de referencia es considerado el mejor ensamblado, ya que se reportó en 36 cromosomas completos, pero datos derivados de un estudio de secuenciación masiva de ARN en *L. major* [7], evidenciaron errores en el ensamblaje del genoma, presumiblemente motivados por el colapso durante el ensamblaje de regiones bordeadas por secuencias repetitivas; dicho estudio, demostró experimentalmente que

siete regiones genómicas habían quedado eliminadas en el proceso de ensamblaje del genoma de *L. major* [8].
El rápido progreso en las tecnologías de secuenciación [9], junto con una reducción significativa de los costes, está facilitando la generación de nuevos ensamblajes genómicos y la mejora de los ya existentes, en los que la determinación precisa del número de genes idénticos dispuestos en tándem, una característica muy común de la organización génica en *Leishmania*, también es fundamental.

Un estudio reciente [10] demuestra que la secuenciación mediante la tecnología de lecturas largas de PacBio (cerca de 20kb) es apropiada para solucionar esta problemática. Por otro lado, la secuenciación de Illumina (que ofrece millones de lecturas, pero de corta longitud), resulta muy relevante y útil para unir con precisión algunos contigs, para extender los extremos cromosómicos y para generar secuencias más fidedignas dado su menor error de secuenciación y profundidad de lectura generada. Así, la combinación de estas dos tecnologías resulta fundamental para lograr un ensamblaje efectivo de genomas de *Leishmania* [10].

Este trabajo pretende obtener el ensamblaje completo, sin fisuras y exento de colapsos en regiones repetidas, del genoma de *L. major*, que pase a ser el genoma modelo de referencia. Este trabajo es novedoso, y combina el uso de distintas tecnologías de NGS y el uso de diversos programas y parámetros. Este ensamblaje proveerá a la comunidad científica de una valiosa información para diversos estudios que abarcan los campos de la genómica, la transcriptómica y la proteómica.

1.2 Objetivos del Trabajo

El objetivo general de este trabajo es:

- Realizar el ensamblaje *de novo* del genoma de *L. major* (cepa Friedlin) a partir de datos de secuenciación obtenidos con las plataformas NGS de Illumina (lecturas cortas) y PacBio (lecturas largas).

Para ello se han utilizado diversas herramientas bioinformáticas y se establecerá un procedimiento que pueda aplicarse de forma general al ensamblaje de genomas de muy diverso origen.

Para la consecución de este objetivo, se han planteado los siguientes objetivos específicos:

1. Realización de un ensamblaje *de novo* del genoma de *L. major* usando lecturas generadas a través de dos tecnologías de secuenciación masiva.
   1.1 Ensamblar de *novo* haciendo uso de lecturas largas de PacBio (285.000 lecturas y 16.000 nucleótidos de longitud media).

1.2 Unir cromosomas fragmentados y ampliar la extensión de los extremos cromosomales usando lecturas cortas de Illumina (cerca de 53 millones de lecturas de 126 nucleótidos de longitud media).
1.3 Validar el ensamblaje realizando una comparación detallada, mediante el manejo de herramientas computacionales, con el genoma de *L. major* que actualmente figura en las bases de datos.
1.4 Identificar regiones genómicas no catalogadas (y presumiblemente nuevos genes) y regiones ordenadas de forma errónea usando herramientas bioinformáticas.

2. Anotación de los genes codificantes de proteínas y los ARN estructurales en el nuevo genoma
   2.1 Realizar la anotación estructural usando herramientas bioinformáticas.
   2.2 Realizar la anotación funcional usando herramientas bioinformáticas.
   2.3 Realizar una curación y revisión manual.

3. Envío de lecturas, genoma y anotación a las bases de datos del ENA y del NCBI.
   3.1 Enviar lecturas de Illlumina a las bases de datos.
   3.2 Enviar lecturas de PacBio a las bases de datos.
   3.3 Enviar la secuencia y anotación del genoma de *L. major* a las bases de datos.

1.3 Enfoque y método seguido

El proceso de ensamblaje del genoma de *L. major* es el pilar fundamental del proyecto. Dependiendo de factores como el material de partida (calidad de lecturas, tipo de lecturas usadas), tipo de ensamblador usado, algoritmos de ensamblaje o parámetros de ensamblaje, el resultado puede variar enormemente. Por ello, en este proyecto, se han planteado tres posibles aproximaciones metodológicas.

Como primera aproximación se planteó ensamblar el genoma de *L. major* solo con lecturas de Illumina, que genera datos de secuenciación con un rendimiento muy alto y un bajo coste. La existencia de algoritmos especializados en ensamblaje de lecturas de Illumina y la precisión (1-2% de errores) [11] de la secuencia obtenida tras el ensamblaje nos llevó a proponer este método como posible estrategia. Sin embargo, a pesar de la baja tasa de error, hay que ser conscientes de que la corta longitud de las lecturas generadas constituye un problema cuando hay repeticiones en tándem en el genoma [12] dado que, durante el ensamblaje, se pueden crear falsas uniones en el genoma en las regiones de repeticiones y generar un ensamblaje genómico con regiones colapsadas, y la consiguiente pérdida de secuencia genómica.

Como segunda opción se planteó hacer un ensamblaje de *novo* con lecturas largas de PacBio que ofrece una longitud de lectura mucho mayor que Illumina, lo que puede ser determinante a la hora de ensamblar de forma correcta fragmentos genómicos con secuencias repetidas en tándem.

Una tercera aproximación estaría basada en un ensamblaje híbrido, usando herramientas que combinan lecturas largas y cortas para la generación de un ensamblaje híbrido. Esta

aproximación es planteada con la finalidad de resolver la principal limitación de las secuencias largas de PacBio, que es su menor precisión en la secuenciación debido a problemas para resolver homopolímeros. Sin embargo, el rápido avance del desarrollo de las tecnologías de secuenciación de tercera generación ha demostrado que los ensamblajes no-híbridos son consistentemente más exitosos [13][14]; si bien, consideramos oportuno también valorar la utilidad de un ensamblaje híbrido para este proyecto.

Asimismo, el uso de lecturas de Illumina se plantea como fundamental para la unión con precisión de contigs, extender los extremos cromosómicos, y la corrección de la secuencia genómica final [15].

Para la comparación detallada del genoma ensamblado con la versión actual de las bases de datos se ha propuesto la generación de scripts en Python y el uso de herramientas bioinformáticas específicas que se irán seleccionando a lo largo del estudio en función de los resultados obtenidos.

En relación con la anotación, se ha propuesto el uso de anotadores *ab initio* que utilizan algoritmos estadísticos para determinar si la secuencia de interés es codificante o no y anotadores comparativos que identifican zonas de alta similitud con genes anotados en organismos relacionados. Para la anotación funcional se usarán distintos métodos dependiendo de los resultados anteriores. La combinación de varios anotadores es primordial para generar una anotación rigurosa y detallada.

Finalmente, las lecturas crudas, los genomas ensamblados y la anotación génica serán depositados en el ENA (European Nucleotide Archive) dado que es una base de datos europea. Una vez se acepten los datos, dicha información también estará disponible en la base de datos del NCBI.

1.4 Planificación del Trabajo

A continuación, se describen las tareas en las que se divide el trabajo, junto con la definición de hitos y objetivos intermedios. Además, se adjunta un diagrama de Gantt con la planificación en el tiempo (Fig.1). También se detalla un análisis de riesgos que pudieran impedir completar el proyecto propuesto.

4.1 Tareas

Las tareas definidas como plan de trabajo inicial son:

a) Ensamblaje *de novo* del genoma de *L. major* (Primera parte)

- Búsqueda de bibliografía: Buscar bibliografía relacionada con abordajes metodológicos y *softwares* específicos para ensamblaje *de novo*.

- Leer bibliografía: Leer tanto la bibliografía recomendada por el tutor relacionada con la biología del parásito como la bibliografía referida en el apartado anterior.

- Utilización de varios ensambladores no-híbridos para lecturas de Illumina: Se procederá a ensamblar las lecturas de Illumina en contigs que, si bien pudieran no servir para generar un ensamblaje completo, podrían ayudar a mejorar el ensamblaje obtenido con lecturas de PacBio y para alargar los extremos de los cromosomas.

- Proceso de ensamblaje con lecturas de PacBio: Llevar a cabo varios ensamblajes no-híbridos de lecturas PacBio modificando parámetros clave del proceso de ensamblaje.

- Proceso de ensamblaje híbrido: Utilizar un ensamblador que use lecturas de PacBio y de Illumina para dar lugar a un ensamblaje híbrido. Se estimará si es la mejor estrategia para alcanzar el objetivo propuesto.

- Identificar y seleccionar los contigs que corresponden a cada cromosoma mediante herramientas bioinformáticas.

- Unir cromosomas fragmentados en varios contigs: en estos casos, se usarán diversas herramientas bioinformáticas para la unión de los contigs que no se han unido en el proceso de ensamblaje y que forman parte del mismo cromosoma. Se determinará a su vez el motivo por el cual se ha producido dicha fragmentación.

- Validar el ensamblaje: Se realizará una comparación detallada con el genoma de *L. major* que actualmente figura en las bases de datos mediante la generación de gráficos de cobertura, y la generación de un pipeline que incluya el uso de herramientas bioinformáticas y la creación de scripts en Python y R.

b) PEC 2 - Redactar resultados (PEC2).

- Redactar los resultados obtenidos hasta la fecha. Se actualizará el plan de trabajo, evaluando los posibles cambios en el mismo.

- Hacer una nueva revisión sintáctica, semántica y ortográfica del escrito.

c) Ensamblaje *de novo* del genoma de *L.major* (Segunda parte)

- Identificar regiones genómicas no catalogadas: Se analizarán zonas de inserción y deleción en el genoma de referencia con respecto al actual, estudiando en detalle aquellos genes afectados en el genoma de referencia y la identificación de nuevos genes en el nuevo ensamblaje.

- Identificar regiones ordenadas de forma errónea: Se analizará mediante herramientas bioinformáticas de visualización aquellas regiones mal ensambladas en el genoma de referencia con respecto a la nueva versión obtenida en este trabajo.

d) Anotación del genoma ensamblado.

- Buscar y leer bibliografía sobre la mejor estrategia a seguir para la anotación de genomas eucariotas.

- Proceder a probar varios anotadores *ab initio* y comparativos, y buscar anotadores que pudieran ser específicos para parásitos relacionados con *Leishmania*.

- Uso de herramientas bioinformáticas para anotación funcional.

- Desarrollar scripts en Python para procesar las anotaciones obtenidas.

- Realizar una curación y revisión manual de los resultados generados comparando con genes ortólogos en otros genomas de *Leishmania*, y se modificarán o eliminarán aquellas anotaciones que pudieran ser erróneas.

e) PEC 3 - Redactar resultados (PEC3)

- Redactar los resultados obtenidos hasta la fecha. Se actualizará el plan de trabajo, evaluando los posibles cambios.

- Realizar una nueva revisión sintáctica, semántica y ortográfica del escrito.

f) Envío de lecturas, genoma y anotación a las bases de datos del ENA y del NCBI.

- Recopilar los datos necesarios para el correcto envío de lecturas de Illumina y PacBio a la base de datos de ENA y se transformará el archivo de anotación en el formato requerido. En este punto, se guardarán y adjuntarán todos los acuses de recibo de ENA que corroboran el correcto envío de los datos.

g) PEC 4- Memoria final (PEC4)

- Incorporar las anteriores PECs a la memoria: Incorporar y adaptar las entregas a la memoria final.

- Redactar de conclusiones: Redactar metodología, justificación de objetivos y conclusiones del trabajo según los resultados obtenidos.

- Revisar de la memoria completa: Revisión sintáctica, semántica y ortográfica de la memoria.

h) PEC 5a) Elaboración de la presentación y autoevaluación

- Seleccionar el contenido a incluir en la presentación.

- Redactar y preparar la presentación con herramienta de diseño tipo PowerPoint o similar, incluyendo gráficas y aspectos relevantes del proyecto teniendo en cuenta el número máximo de transparencias y tiempo de exposición.

- Realizar una revisión exhaustiva de la presentación: revisión sintáctica, semántica y ortográfica.

- Realizar un informe de autoevaluación como ejercicio de autocrítica y una estimación del cumplimiento de la planificación. Rellenar informe con datos.

i) PEC 5b) Defensa pública

-Revisar la memoria del trabajo para presentarla y defenderla ante un tribunal.

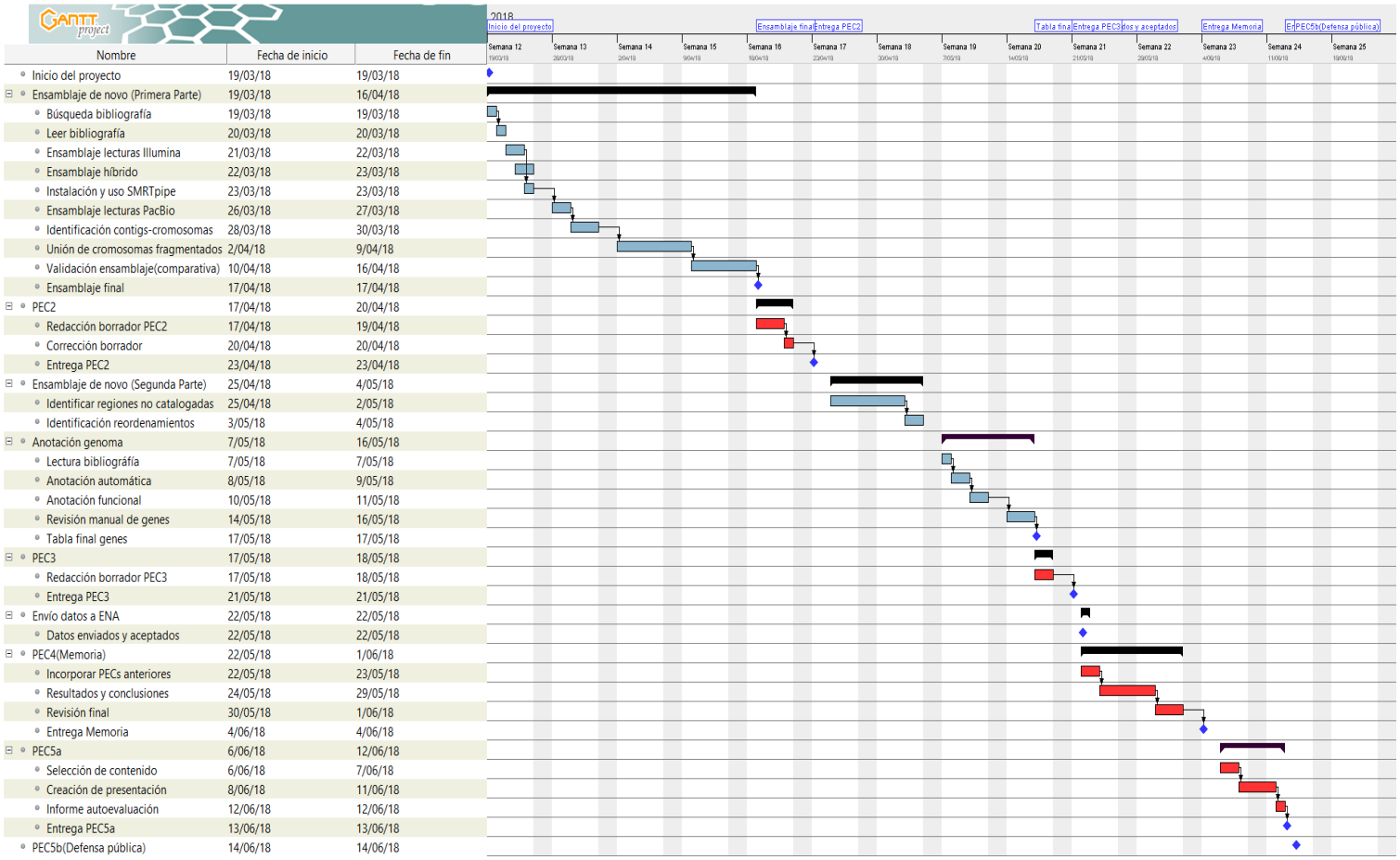

Figura 1. Diagrama Gantt. Planificación del proyecto desde el inicio hasta el final del proceso.
Las principales tareas se marcan en negro. Los hitos están señalados en azul oscuro. En rojo se marcan las redacciones entregables.

4.2 Hitos del TFM

En la siguiente tabla se muestran los hitos del trabajo que se corresponden con las entregas programadas según el Plan Docente (en verde), y los objetivos marcados en el trabajo.

Tabla 1. Hitos del TFM

Hitos	**Fechas**
Inicio del proyecto	19 de marzo
Ensamblaje final	17 de abril
PEC2	23 de abril
Anotación del genoma	17 de mayo
PEC3	21 de mayo
Envío datos a ENA	22 de mayo
PEC4(Memoria Final)	4 de junio
PEC5a	13 de junio
PEC5b (Defensa pública)	14– 25 de junio

4.3 Análisis de riesgo

A continuación, se detallan las posibles incidencias que pueden surgir a lo largo de la realización del TFM.

- Retraso en la instalación de los programas requeridos. Se pedirá ayuda al administrador de sistemas del CBMSO si fuese necesario.

- Avería del clúster donde se harán los cálculos y procesos.

- Incompatibilidad del ordenador para realizar los grandes requerimientos computacionales que conlleva el ensamblaje de genomas. Se podrían realizar en centros de cálculo con gran capacidad (Centro de Computación Científica (CCC) y Red Española de Supercomputación (RES).

- Necesidad de buscar nuevas herramientas computacionales con algoritmos diferentes a los propuestos para la mejora de resultados.

- Falta de tiempo para la realización de todas las tareas. En este caso, se priorizarán las tareas más importantes y se redactarán líneas de trabajo futuras incluyendo las tareas pendientes.

1.5 Breve sumario de productos obtenidos

Los productos obtenidos a la finalización del trabajo son:
- Memoria: Escrito de la memoria final incluyendo todos los objetivos, metodología, resultados y conclusiones obtenidos.
- Presentación virtual: Presentación que servirá de apoyo a la memoria que se defenderá ante un tribunal.
- Creación de un ensamblaje completo sin fisuras y exento de colapsos en regiones repetidas del genoma de *L. major*.
- Obtención de un archivo de anotación para los genes presentes en el genoma de *L.major*.
- Obtención de una lista de genes afectados en el genoma de referencia por grandes inserciones o deleciones en el nuevo ensamblaje.
- Desarrollo de scripts en Python, R y Gnuplot para la validación del ensamblaje, generación de figuras y curación de ficheros de anotación.

1.6 Breve descripción de los otros capítulos de la memoria

La memoria final del trabajo consta de los siguientes capítulos:

- Ensamblaje *de novo* del genoma de *L. major*. Se detalla todo el proceso llevado a cabo para el ensamblaje del genoma del parásito, así como el proceso de validación del ensamblaje mediante la comparación detallada frente al genoma de referencia. El capítulo finaliza con la identificación de regiones genómicas no catalogadas y la de regiones ordenadas de forma errónea.

- Anotación de los genes codificantes de proteínas y los ARN estructurales en el nuevo genoma. Se describe el programa bioinformático usado y se mostrarán los scripts generados para la revisión y validación tras el proceso de anotación.

- Envío de lecturas de secuenciación a las bases de datos del ENA y del NCBI. Se explica el proceso llevado a cabo para el envío de lecturas en la base de datos ENA. Se indicará el número de proyecto asignado para acceder a estos datos una vez sean públicos.

- Conclusiones: Se presentan las conclusiones del trabajo, se analiza el seguimiento de la planificación original y, por último, se plantean las posibles líneas futuras de trabajo.

- Glosario: Definición de términos y acrónimos empleados en el presente documento.

- Bibliografía: Se muestra información sobre la bibliografía utilizada en la elaboración del trabajo.
- Anexos: Contiene código de algunos scripts generados y recibos de envío de secuencias a las bases de datos.

2. Ensamblaje *de novo* del genoma de *Leishmania major*.

El primer paso antes de realizar cualquier ensamblaje o estudio de secuenciación masiva es analizar las calidades de las lecturas y conocer la naturaleza y características de estas.

Material de partida

Lecturas Illumina.

La construcción de la librería y la secuenciación (paired-end) se llevó a cabo en el Centro Nacional de Análisis Genómico (CNAG-CRG, España) utilizando la tecnología Illumina HiSeq 2000. Se generaron un total de 52,845,525 lecturas pareadas de 126 nt. Las lecturas fueron filtradas por calidad (valor de 20) y se seleccionaron las lecturas con longitud ≥60-nt.

Lecturas PacBio.

Para la generación de lecturas largas se utilizó la tecnología de secuenciación de molécula única en tiempo real (SMRT) desarrollada por Pacific Biosciences (PacBio) [16]. Se generaron un total de 285,082 lecturas pre-filtradas usando el secuenciador PacBio RS II. El servicio que llevó a cabo la secuenciación fue el Norwegian Sequencing Center [17] una plataforma tecnológica nacional organizada por la Universidad de Oslo. El filtrado de calidad de las lecturas de PacBio fue llevado a cabo durante el propio ensamblaje.

2.1 Utilización de ensambladores no-híbridos para lecturas de Illumina.

Ensamblaje de las lecturas de Illumina con la finalidad de poder alargar extremos de cromosomas.

En base a una revisión bibliográfica sobre la metodología bioinformática empleada en el ensamblaje de genomas y teniendo en cuenta las características del organismo a estudiar (*L. major*, cepa Friedlin), se seleccionaron dos ensambladores no híbridos de lecturas de Illumina: CLC-bio y SPAdes.

Herramientas bioinformáticas

- CLC-bio

CLC-bio [18] version 5.0 es un software propiedad de QIAGEN [19] que permite el análisis de datos de secuenciación de las principales plataformas mediante el uso de varias aplicaciones. Uno de los objetivos del uso de este software era probar las aplicaciones relacionadas con el ensamblaje de genomas y comparar los resultados con otras herramientas de libre acceso. El comando usado para llevar a cabo el ensamblaje fue:

```
$clc_assembler -o <scaffolds.fasta> --paired fb ss 180 310 --reads
<forware.fastq> <reverse.fastq>
```

donde:

`--paired fb: Indica que se dispone de lecturas pareadas.`

`ss: tamaño aproximado de inserto.`

- SPAdes

SPAdes [20] es una herramienta de acceso libre para el ensamblaje de genomas basado en la construcción de grafos *de Bruijn* [21] que mide la relación que existe entre las sub-cadenas de nucleótidos de longitud fija (*k-mer*) creadas y genera un grafo donde los nodos son los *k-mers* y las conexiones del grafo indican que los *k-mers* son adyacentes y solapan (*k -1* nucleótidos) lo que tiene una gran ventaja en cuanto al coste del tiempo computacional. Evidencias bibliográficas muestran a SPAdes como uno de los mejores ensambladores que existen actualmente para el ensamblaje de lecturas de Illumina debido a su calidad [22], alta precisión [23] y a la utilidad para corregir errores de secuencia.

Debido al hecho de que SPAdes trabaja con *k-mers*, se decidió usar KmerGenie [24] que eligió el *k-mer* 53 como el más adecuado para hacer el ensamblaje. Sin embargo, dado que SPAdes hace iteraciones de *k-mers*, se decidieron usar varios. El comando usado fue:

```
$ spades.py -1 <forware.fastq> -2 reverse.fastq --careful -o spadesReadsIllumina -k 21,33,53,77,99,125
```

Comparación de métodos

En primer lugar, se seleccionaron dos programas específicos de ensamblaje de lecturas Illumina para llevar a cabo una comparación de los resultados obtenidos y, en caso de que fuera necesario, alargar extremos de cromosomas. En la tabla 1 se muestran las diferencias entre los diferentes resultados obtenidos.

Tabla 2. **Características generales de ensamblajes de Illumina.**

Ensamblaje	contigs	scaffolds	Bases totales (pb)	Gaps	Ns	N50	Long. Máx.(pb)	Long. Min.(pb)
CLC-bio	-	2,182	30,505,152	1138	60,156	33,407	284,279	466
SPAdes	-	2,307	31,123,422	265	2,383	37,528	174,832	126

Como puede observarse en la tabla, a pesar de que el ensamblaje de SPAdes contiene menos scaffolds, el número y tamaño de los gaps es menor que en el ensamblaje de CLC y el N50 (longitud mínima del contig que se necesita para cubrir el 50% de la longitud total del genoma) es más alto. Con estos resultados, el ensamblaje completo del genoma en 36 cromosomas y sin gaps no fue obtenido utilizando lecturas de Illumina. En definitiva, estos ensamblajes son comparables y aunque en pasos posteriores ambos pueden ser útiles, SPAdes, como herramienta gratuita, podría ser el programa de elección.

2.2 Ensamblaje de lecturas PacBio.

Ensamblajes no-híbridos de lecturas PacBio modificando parámetros clave del proceso de ensamblaje.

El proceso de ensamblaje del genoma con lecturas largas de PacBio se llevó a cabo con HGAP [25]. El proceso en sí mismo se basa en una sucesión de pasos para generar ensamblajes *de novo.* En primer lugar, se seleccionan las secuencias más largas y precisas y se genera una secuencia consenso mapeandosobre ellas el resto de las lecturas.

Herramientas bioinformáticas

El proceso de ensamblaje usa un algoritmo llamado Overlap Layour Consensus (OLC) [26] que identifica pares de lecturas que solapan correctamente, donde cada lectura se representa gráficamente como un nodo y las superposiciones se representan como conexiones que unen los dos nodos implicados. Finalmente, corrige el ensamblaje final generando una secuencia consenso.

Con el objetivo de obtener un ensamblaje mejorado, se llevaron a cabo varios ensamblajes con HGAP variando parámetros y versiones del programa. Los protocolos de HGAP usados fueron Pacific Biosciences SMRT Analysis Software v2.3.0 (HGAP3) y SMRT Link 4.0.0 (HGAP4) incluidos en SMRTPipe que se instaló previamente. Se realizaron un total de 3 ensamblajes (estimando el tamaño de genoma esperado en HGAP3 con 35 y 33 Mb, HGAP4 en 35 Mb). Las principales características de los resultados obtenidos se muestran en la Tabla 3.

Tabla 3. Características generales de ensamblajes con lecturas PacBio.

Ensamblaje	contigs	scaffolds	Bases totales (pb)	Gaps	Ns	N50	Long. Máx.(pb)	Long. Min.(pb)
HGAP3(35)	84	0	33,440,577	0	0	833,325	2,756,814	715
HGAP3(33)	97	0	33,515,146	0	0	872,715	2,748,247	938
HGAP4(35)	81	0	32,690,047	0	0	750,558	2,248,458	1,916

En la tabla, puede observarse que los tres ensamblajes realizados con lecturas de PacBio son comparables y muy similares. Cabe destacar que a pesar de que el segundo intento de ensamblaje con HGAP3 (HGAP3_33) genera un número mayor de contigs, el valor de N50 es mayor a los anteriores y el número de bases totales también es mayor. La generación de varios ensamblajes permite seleccionar los contigs mejor ensamblados para formar los cromosomas.

2.3 Proceso de ensamblaje híbrido.

Uso de ensamblador que use lecturas de PacBio y de Illumina para dar lugar a un ensamblaje híbrido. Estimación para ver la mejor estrategia para llevar a cabo el proceso posterior.

La combinación de las tecnologías de secuenciación de lecturas largas y lecturas cortas ofrece ventajas competitivas para los proyectos de secuenciación de genomas eucarióticos [27]. Por un lado, el uso exclusivo de tecnologías de secuenciación de segunda generación (como las lecturas de Illumina) para el ensamblaje del genoma, puede fallar o conducir al ensamblaje incompleto de zonas importantes del genoma. Por otro lado, la principal limitación de las tecnologías de tercera generación (como PacBio) es su precisión relativamente baja, causando errores en el ADN secuenciado si no se dispone de una cobertura adecuada [28], si bien, los avances en la tecnología de secuenciación van desarrollando sistemas que producen lecturas largas con fidelidad cada vez más alta. La disminución del coste, el aumento de la eficiencia y la mejora del rendimiento de la generación de datos de secuenciación de tercera generación está cambiando las estrategias de trabajo, y los programas y algoritmos de secuenciación están cada vez más optimizados para realizar ensamblajes no-híbridos.

El principal objetivo de este punto es conocer las diversas opciones que existen actualmente para llevar a cabo el proceso de ensamblaje cuando se dispone de datos que provienen de varias tecnologías de secuenciación. Posteriormente, se compararán los resultados obtenidos mediante los ensamblajes no-híbridos que han llevado a cabo en este trabajo.

Herramientas bioinformáticas

- SPAdes-hybrid

Las principales tareas del ensamblador híbrido de SPAdes [29] son construir el grafo de ensamblaje a partir de lecturas cortas (Illumina) usando SPAdes y acortar la distancia entre grafos usando la secuencia consenso de lecturas largas (PacBio), permitiendo una mejora en el ensamblaje en regiones con repeticiones genómicas. Los comandos usados fueron:

```
$ spades.py --pacbio corrected.fastq -1 forware.fastq -2 reverse.fastq --careful -o spadesReadsIlluminaReadsPacBio -k 53,77,99,125
```

- AHA: A Hybrid Assembler

El ensamblador híbrido AHA [30] combina lecturas largas (generadas por PacBio) con ensamblajes de lecturas cortas (contigs generados con SPAdes a partir de lecturas Illumina), de forma que, usando las lecturas de PacBio, ordena a los contigs de "alta confianza" uniéndolos en contigs más grandes o scaffolds.

- IDBA_hybrid.

Por último, se propuso realizar un ensamblaje híbrido con IDBA-hybrid [31], que podría ofrecer buenos resultados a pesar de que suele usarse para ensamblaje *de novo* cuando la profundidad de secuenciación es baja y el genoma de referencia es similar al genoma objetivo. IDBA-Hybrid es un ensamblador iterativo que usa grafos *de Bruijn*, cuyo objetivo es utilizar un genoma de referencia (contigs de Illumina) para ayudar al

ensamblaje *de novo*. En este caso, IDBA-hybrid alinea las lecturas de PacBio frente a los contigs ensamblados de Illumina (ensamblados con SPAdes), corrige las regiones similares alineadas y finalmente, agrupa todas las lecturas y los contigs obtenidos de ese alineamiento para hacer ensamblaje *de novo*. El comando usado para llevar a cabo el ensamblaje fue:

```
$ idba_hybrid -r <contigsSpades.fasta> --long_read corrected.fasta  --
mink 20 --maxk  124 -o output_file
```

donde:

```
--mink : numero minimo de k-mero a usar
--maxk : numero maximo de k-mero a usar
```

Comparación de métodos híbridos.

Los ensamblajes de alta calidad y bajo costo son importantes para anotar de forma precisa los genomas. El ensamblaje híbrido de lecturas cortas y largas [32], ha resultado útil en muchos proyectos de secuenciación. Por ello, un objetivo clave de este estudio era comprobar la efectividad de los ensamblajes híbridos automáticos en este tipo de proyectos.

Una vez finalizado el proceso de ensamblaje, se realizó una comparativa entre los diferentes ensamblajes llevados a cabo. En la tabla 4 se muestran las características generales de cada ensamblaje.

Tabla 4. Características generales de ensamblajes híbridos.

Ensamblaje	contigs	scaffolds	Bases totales (pb)	Gaps	Ns	N50	Long. Máx.(pb)	Long. Min.(pb)
SPAdes-hybrid	-	1,681	31,709,286	174	1,456	53,467	249,263	126
AHA	-	644	31,906,639	3,128	1,955,221	26,6754	1,178,335	466
IDBA-hybrid	-	46,746	38,608,021	2	43	10,772	82,478	124

Como puede observarse en la tabla, IDBA-hybrid genera los peores resultados de ensamblaje, dado que genera una cantidad de scaffolds mucho mayor en comparación con el resto y un N50 menor. Por otro lado, AHA, es capaz de reducir el número de scaffolds de Illumina, sin embargo, lo hace a costa de incrementar el número de gaps. SPAdes-hybrid reduce el número de scaffolds disminuyendo considerablemente el número de gaps y aumentando el N50.

Tras este proceso, se ha podido concluir que, a pesar de que los ensamblajes híbridos pueden ayudar de forma significativa a mejorar el ensamblaje realizado con lecturas de Illumina, los resultados se alejan mucho del objetivo principal de obtener un ensamblaje completo, sin fisuras y exento de colapsos.

2.4 Comparación de ensamblajes generados.

Estrategias llevadas a cabo con la finalidad de estimar el mejor flujo de trabajo que se debe seguir.

Habiendo realizado diversos ensamblajes con diferentes estrategias y parámetros, en este punto se pretende estimar la mejor estrategia a seguir para llevar a cabo ensamblajes de organismos con similares características. Para ello, se usó la herramienta QUAST [33], que evalúa la calidad de los ensamblajes y selecciona el mejor ensamblaje de todos los proporcionados. En las Figuras 2 y 3 se muestran los resultados de la comparativa de QUAST de los ensamblajes realizados. El comando utilizado para llevar a cabo esta comparación fue el siguiente:

```
$ python quast.py <lista_ensamblajes>
```

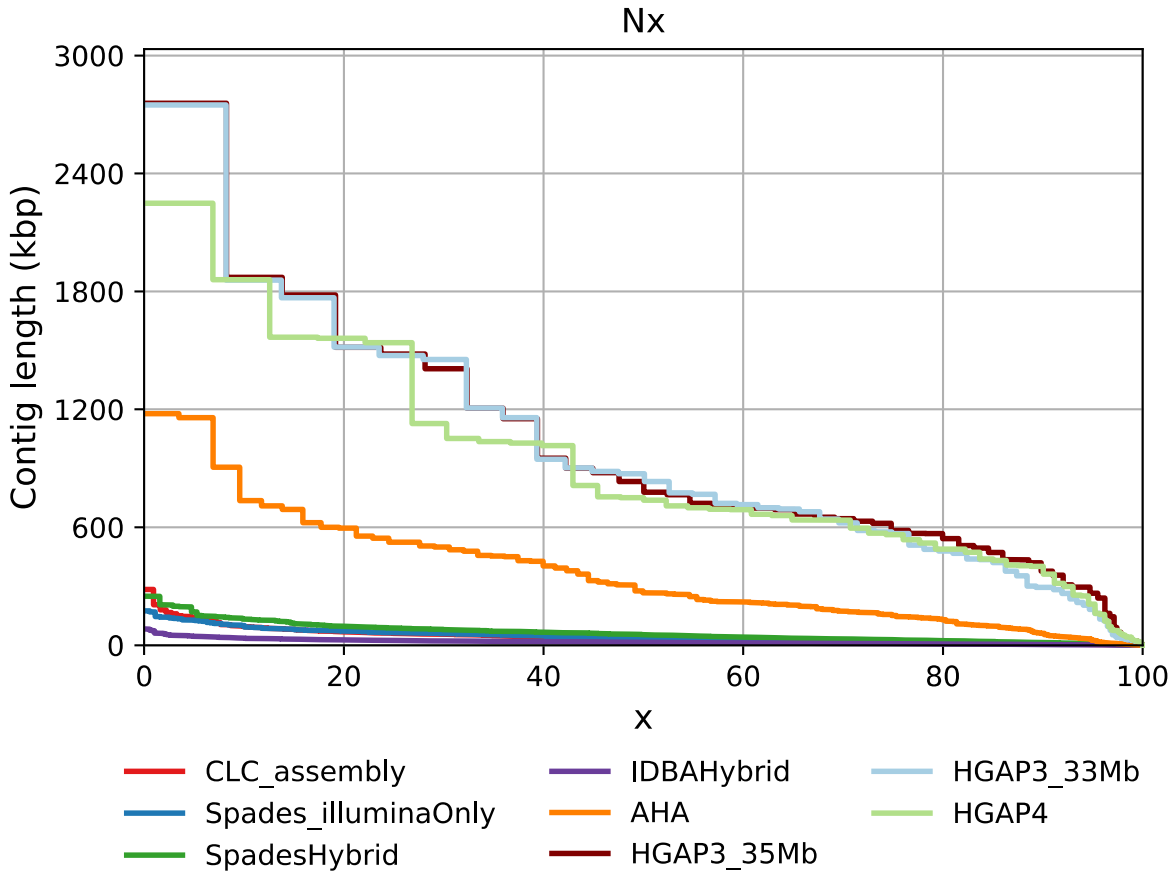

Figura 2. Gráfico comparativo generado por QUAST.
Muestra el crecimiento de las longitudes de los ensamblajes en función a sus valores de Nx (N50, N75, etc). El eje x varía de 0 a 100 para cada valor de Nx. El eje Y representa las longitudes de los ensamblajes. Los ensamblajes con mayor tamaño de contig son los obtenidos con lecturas de PacBio (HGAP3_35Mb, HGAP3_33Mb y HGAP4). Los ensamblajes de lecturas de illumina (CLC_assembly y Spades_illuminaOnly) e híbridos (IDBAHybrid y SpadesHybrid) mostraron peores resultados.

Worst Median Best ☑ Show heatmap

Statistics without reference	CLC_assembly	Spades_illuminaOnly	SpadesHybrid	IDBAHybrid	AHA	HGAP3_35Mb	HGAP3_33Mb	HGAP4
# contigs	2181	1871	1331	4172	643	84	97	81
# contigs (>= 0 bp)	2182	2307	1681	46 746	644	84	97	81
# contigs (>= 1000 bp)	1783	1576	1116	3282	445	82	96	81
# contigs (>= 5000 bp)	1160	1117	885	1817	274	76	86	76
# contigs (>= 10000 bp)	837	811	717	1041	210	74	84	74
# contigs (>= 25000 bp)	380	423	423	202	166	56	67	61
# contigs (>= 50000 bp)	136	151	200	17	126	47	50	53
Largest contig	284 279	174 832	249 263	82 478	1 178 335	2 756 814	2 748 247	2 248 458
Total length	30 564 842	30 979 596	31 596 890	30 236 816	33 861 394	33 440 577	33 515 146	32 690 047
Total length (>= 0 bp)	30 565 308	31 125 805	31 710 742	38 608 064	33 861 860	33 440 577	33 515 146	32 690 047
Total length (>= 1000 bp)	30 273 766	30 758 507	31 434 784	29 605 040	33 718 380	33 439 004	33 514 208	32 690 047
Total length (>= 5000 bp)	28 682 094	29 574 363	30 826 587	25 796 028	33 322 425	33 420 719	33 490 774	32 673 868
Total length (>= 10000 bp)	26 349 158	27 324 569	29 592 547	20 200 923	32 852 885	33 404 654	33 473 640	32 660 570
Total length (>= 25000 bp)	18 895 694	20 957 719	24 697 867	7 092 145	32 221 274	33 066 449	33 164 721	32 416 011
Total length (>= 50000 bp)	10 516 540	11 500 759	16 917 451	1 035 448	30 687 177	32 761 164	32 608 832	32 119 615
N50	33 407	37 625	53 735	14 849	266 754	833 325	872 715	750 558
N75	16 278	19 365	28 097	7775	151 372	584 298	568 491	562 465
L50	256	244	179	622	33	12	12	13
L75	576	526	385	1321	74	25	25	26
GC (%)	59.28	59.36	59.4	59.25	59.22	59.75	59.8	59.86
Mismatches								
# N's	60 154	2383	1456	43	1 955 219	0	0	0
# N's per 100 kbp	196.81	7.69	4.61	0.14	5774.18	0	0	0

Figura 3 Resultados estadísticos de la comparación de ensamblajes realizada con QUAST.

Los ensamblajes no-híbridos con lecturas de PacBio muestran mejores resultados que los llevados a cabo con lecturas Illumina o con ensamblajes híbridos. Como puede observarse en la figura 3, los resultados se colorean en azul o rojo dependiendo si sus resultados se consideran buenos o malos respectivamente. Los ensamblajes de Illumina muestran los peores resultados, seguidos de los ensamblajes híbridos. A pesar de que IDBAHybrid muestra buenos resultados en cuanto a longitud total, el resto de los valores son mejorables. Sin embargo, los ensamblajes llevados a cabo con las lecturas de PacBio, muestran en general valores aceptables y mejores en comparación con los demás métodos.

Teniendo estos resultados en cuenta, aquellos estudios que cuenten con lecturas de Illumina, pero baja cobertura de PacBio, el ensamblaje híbrido puede ser una alternativa factible [34], pero no es del todo necesario cuando se dispone de alta cobertura en lecturas largas. Sin embargo, cabe destacar que estas comparaciones no tienen en cuenta posibles errores de secuenciación. Este hecho será analizado más adelante.

2.5 Identificación y selección de contigs que corresponden a cada cromosoma mediante herramientas bioinformáticas.

Después de seleccionar el ensamblaje no-híbrido de PacBio como mejor estrategia a seguir, se procedió a identificar la correspondencia de los contigs obtenidos y los 36 cromosomas de *L. major*.

A partir de una evaluación inicial de ensamblajes y coberturas, los contigs con una cobertura inusualmente baja (<40x) y de corta longitud (<15kb) se descartaron y se consideraron contigs espurios. Un análisis posterior de estos contigs adicionales con BLAST [35] y el alineador múltiple MAFFT [36] concluyó que estos contigs tenían un alto porcentaje de similitud con “contigs cromosómicos” que, además de ser más largos, tienen una cobertura razonablemente uniforme. La Figura 4 muestra un ejemplo de este comportamiento en el cromosoma 36. Las identidades de secuencia de los contigs evaluadas a través de BLAST utilizando como referencia los cromosomas del genoma de

referencia de *L. major* (cepa Friedlin) [8] se llevaron a cabo mediante los siguientes comandos:

```
$ makeblastdb -in <referencia> -dbtype nucl -out database
```

```
$blastn-query <major_denovoassembly> -db database -outfmt 6-
max_target_seqs 1> outputFiletxt
```

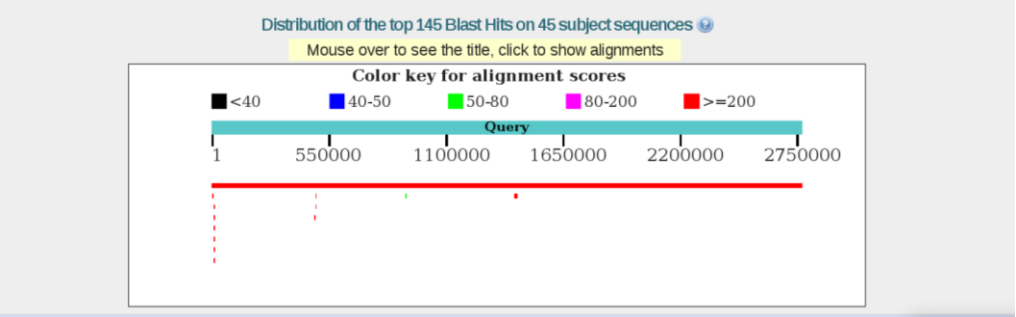

Figura 4 Alineamiento de BLAST.
Los contigs de HGAP se alinearon frente al chromosoma 36 de *Leishmania major* Friedlin. El contig más largo que posee cobertura alta (línea roja) representa todo el cromosoma, mientras que los contigs pequeños con baja cobertura (puntos en rojo) fueron descartados.

Después del filtrado, se seleccionaron un total de 41 contigs que forman los cromosomas de *L.major* Friedlin. El cierre total del genoma generalmente depende de la capacidad de generar una cobertura suficiente con lecturas de longitud muy larga que alineen a ambos lados de regiones con repetición interna.

2.6 Análisis del motivo por el cual se produce fragmentación de cromosomas.

En este punto, se analizó en detalle el motivo por el cual, en ciertos casos, a pesar de disponer de alta cobertura, el ensamblador no es capaz de generar un cromosoma continuo y lo fragmenta. Los cromosomas fragmentados se alinearon frente a sí mismos con Gepard [37], que genera un diagrama de puntos para cada comparación. En las figuras 5-9 se muestran los resultados de los dotplots para cada cromosoma.

Las proteínas idénticas obviamente tendrán una línea diagonal en el centro de la matriz. Las inserciones y eliminaciones entre secuencias dan lugar a interrupciones en esta diagonal. Las regiones de similitud local o secuencias repetitivas dan lugar a coincidencias diagonales adicionales además de la diagonal central.

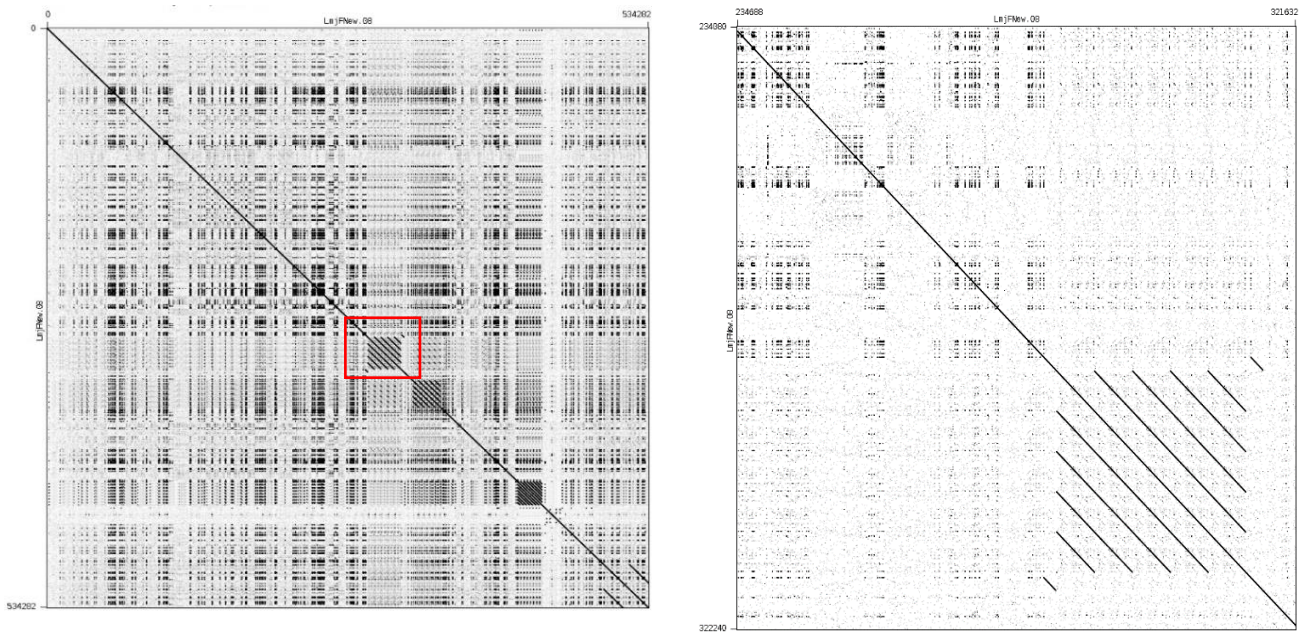

Figura 5 Dotplot. Alineamiento del ensamblado del cromosoma 8 frente a sí mismo mediante Gepard. En los diagramas de puntos, las regiones idénticas tendrán una línea diagonal en el centro de la matriz. Las inserciones y deleciones entre secuencias dan lugar a interrupciones en esta diagonal y las regiones repetitivas dan lugar a coincidencias diagonales adicionales además de la diagonal central. En la izquierda, se muestra el alineamiento completo. El diagrama de la derecha corresponde a la ampliación de la zona marcada en rojo (izquierda) que corresponde con zona repetitiva donde se produce la rotura del cromosoma.

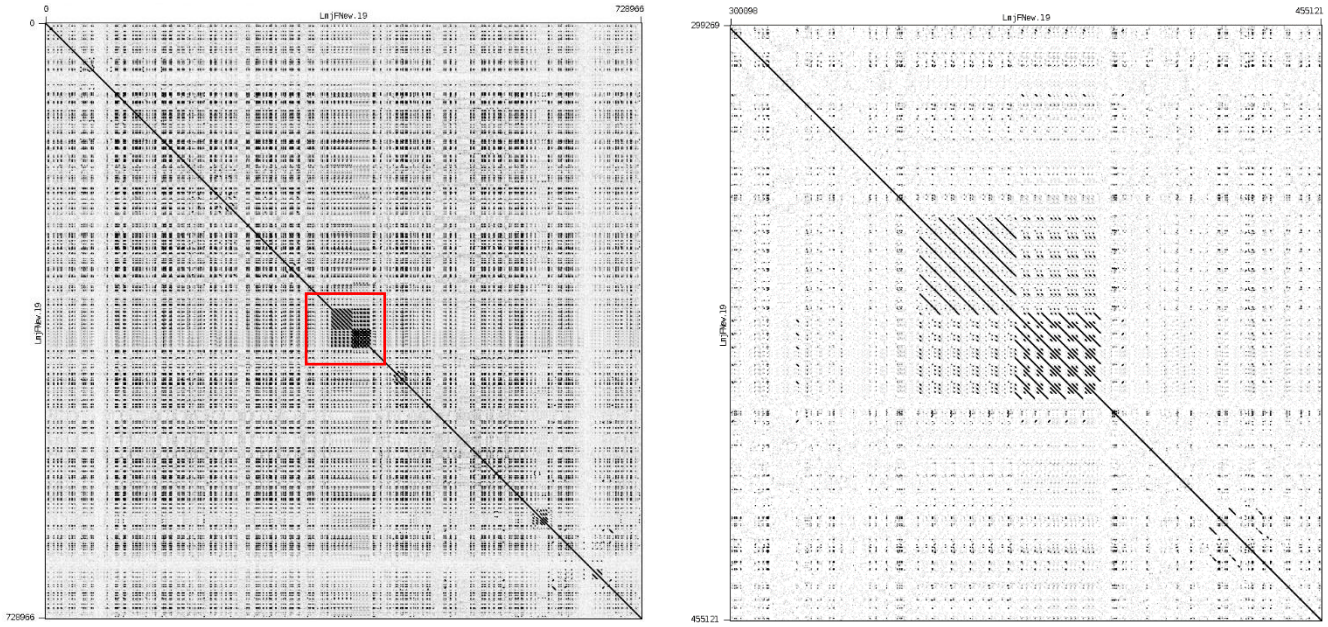

Figura 6 Dotplot. Alineamiento del ensamblado del cromosoma 19 frente a sí mismo mediante Gepard En los diagramas de puntos, las regiones idénticas tendrán una línea diagonal en el centro de la matriz. Las inserciones y deleciones entre secuencias dan lugar a interrupciones en esta diagonal y las regiones repetitivas dan lugar a coincidencias diagonales adicionales además de la diagonal central. En la izquierda, se muestra el alineamiento completo. El diagrama de la derecha corresponde a la ampliación de la zona marcada en rojo (izquierda) que corresponde con zona repetitiva donde se produce la rotura del cromosoma.

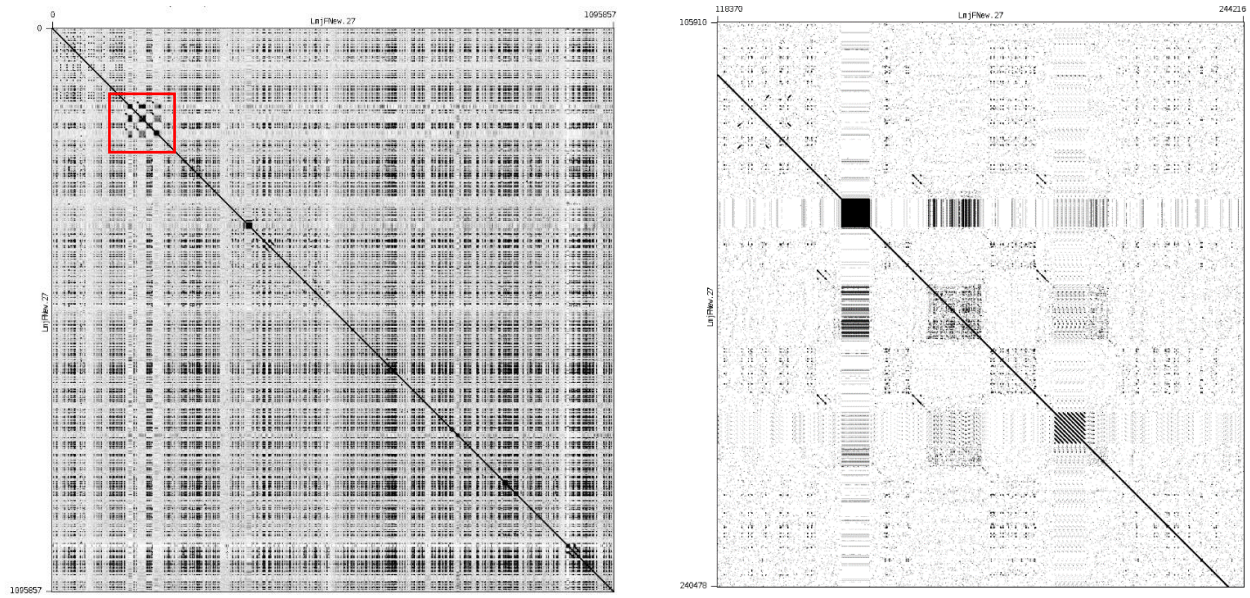

Figura 7 Dotplot. Alineamiento del ensamblado del cromosoma 27 frente a sí mismo mediante Gepard. En los diagramas de puntos, las regiones idénticas tendrán una línea diagonal en el centro de la matriz. Las inserciones y deleciones entre secuencias dan lugar a interrupciones en esta diagonal y las regiones repetitivas dan lugar a coincidencias diagonales adicionales además de la diagonal central. En la izquierda, se muestra el alineamiento completo. El diagrama de la derecha corresponde a la ampliación de la zona marcada en rojo (izquierda) que corresponde con zona repetitiva donde se produce la rotura del cromosoma.

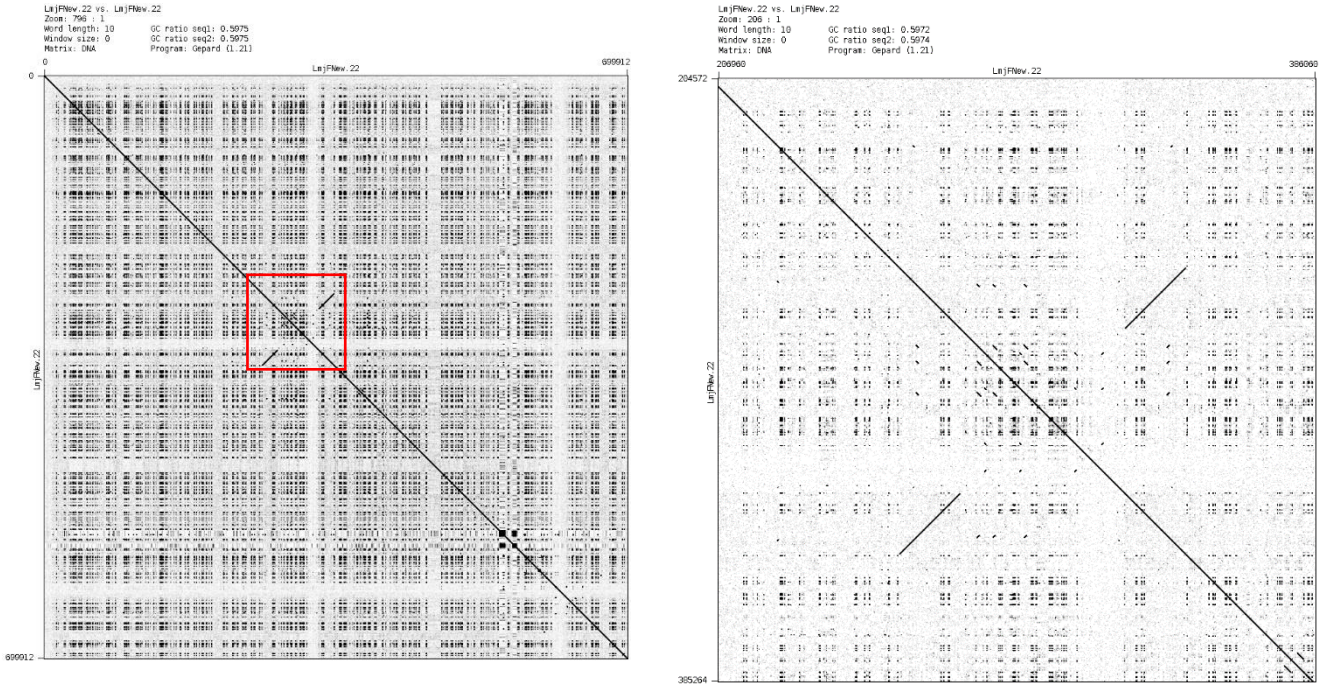

Figura 8 Dotplot. Alineamiento del ensamblado del cromosoma 22 frente a sí mismo mediante Gepard. En los diagramas de puntos, las regiones idénticas tendrán una línea diagonal en el centro de la matriz. Las inserciones y deleciones entre secuencias dan lugar a interrupciones en esta diagonal y las regiones repetitivas dan lugar a coincidencias diagonales adicionales además de la diagonal central. En la izquierda, se muestra el alineamiento completo. El diagrama de la derecha corresponde a la ampliación de la zona marcada en rojo (izquierda) que corresponde con zona repetitiva donde se produce la rotura del cromosoma.

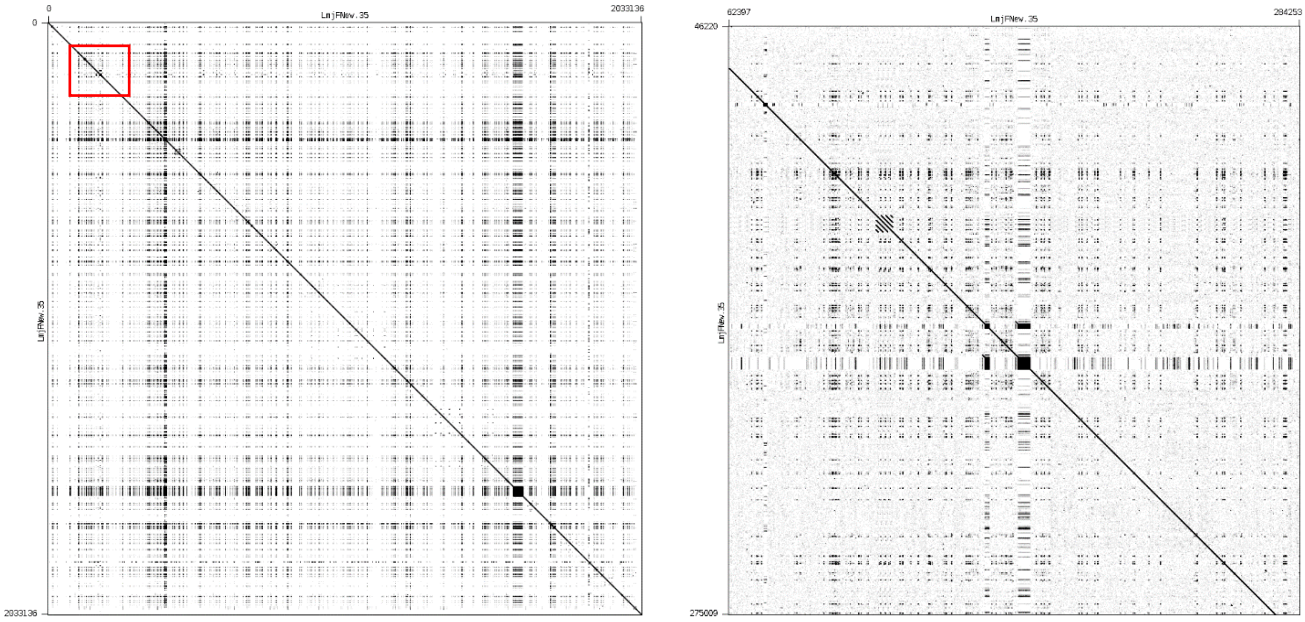

Figura 9 Dotplot. Alineamiento del ensamblado del cromosoma 35 frente a sí mismo mediante Gepard. En los diagramas de puntos, las regiones idénticas tendrán una línea diagonal en el centro de la matriz. Las inserciones y deleciones entre secuencias dan lugar a interrupciones en esta diagonal y las regiones repetitivas dan lugar a coincidencias diagonales adicionales además de la diagonal central. En la izquierda, se muestra el alineamiento completo. El diagrama de la derecha corresponde a la ampliación de la zona marcada en rojo (izquierda) que corresponde con zona repetitiva donde se produce la rotura del cromosoma.

Dado que la longitud media de las lecturas de PacBio obtenidas es de 16 kb aproximadamente y que, tras observar las figuras, la fragmentación de los cromosomas tiene lugar en zonas altamente repetitivas de entre 20 a 60 kb de longitud, se puede concluir que el tamaño de la unidad de repetición justifica que las lecturas de PacBio también colapsen durante el ensamblaje. Por otro lado, es posible que existan ciertas zonas del genoma cuya secuenciación se vea afectada por las características de su secuencia de ADN en este organismo, ya sea por su alto contenido en GC, regiones de cambio de hebra transcripcional o la existencia de un gran número de estructuras G-cuádruplex. En el cromosoma 22, a pesar de que la rotura se localiza en una zona aparentemente libre de repeticiones, la existencia de dos largas repeticiones invertidas podría ser el motivo de la fragmentación.

2.7 Unión de cromosomas fragmentados en varios contigs. Herramientas bioinformáticas para la unión de los contigs que no se han unido en el proceso de ensamblaje y que forman parte del mismo cromosoma.

Un total de cinco cromosomas se obtuvieron en dos contigs cada uno tras el proceso de ensamblaje. En este punto se usaron varias herramientas especializadas en unir contigs mediante diversas estrategias. Finalmente, todos los contigs (cromosomas 8, 19, 22, 27 y 35) se unieron por minimus2 [38] que usa nucmer [39] para calcular los solapamientos entre contigs y unir a través de la región de solapamiento. Los comandos utilizados fueron:

```
$ toAmos -s <fasta> -o out.afg
$ minimus2 <input>
```

2.8 Validación del ensamblaje.

Comparación detallada con el genoma de *L. major* que actualmente figura en las bases de datos mediante la generación de gráficos de cobertura. Extensión de extremos cromosómicos con la ayuda de lecturas de Illumina.

Tras la generación de un ensamblaje preliminar, se procedió a validar la calidad de dicho ensamblaje comparando frente al genoma de referencia.

Los cromosomas del nuevo ensamblaje se alinearon frente a los cromosomas de referencia utilizando el alineador LAST [40]. Los comandos usados fueron:

```
$ lastdb -cR11 -uNEAR db archivo.fasta
$ lastal -m50 -E0.05 db <archivo.fasta> | last-split -m1 > archivo.maf
$ maf-convert sam archivo.maf > archivo.sam
```

Tras el alineamiento se observó que alguno de los cromosomas ensamblados, tenían extremos más cortos que los cromosomas del genoma *L.major* de referencia.

Para averiguar si la secuencia que faltaba se debía a problemas a la hora de ensamblar extremos de cromosomas con lecturas de PacBio o se trataba de artefactos en el genoma de referencia de éste parásito, se usó de nuevo LAST para alinear los contigs de Illumina frente al genoma de referencia. Esto permitió la identificación de contigs de Illumina que alineaban con los extremos cromosómicos pero que tenían secuencias que sobresalían. De esta forma, se confirmó que el ensamblaje de lecturas de PacBio no consigue ensamblar los extremos cromosomales y se procedió a diseñar una estrategia para extender dicha secuencia mediante lecturas y ensamblaje de Illumina.

Finalmente, se usaron varias herramientas para extender algunos extremos cromosómicos [10] tales como MAFFT o minimus2. Los gaps generados tras la extensión de los contigs se cerraron con GapFiller [41] y GapCloser [42]. Ambos programas usan lecturas de Illumina para cerrar los gaps.

Finalmente, se llevó a cabo la extensión de los extremos del resto de cromosomas mediante SSPACE-standard [43]. En el anexo I se muestra la lista de cromosomas extendidos. El comando usado fue:

```
$ perl SSPACE_Basic_v2.0.pl -l library.txt -s <assembly> -k 5 -a 0.7
-x 1 -b <out>
```

Posteriormente, se llevó a cabo un estudio detallado de la calidad del ensamblaje frente a los dos tipos de lecturas empleados, PacBio e Illumina, observándose de acuerdo con el alineamiento de lecturas Illumina, que en el genoma ensamblado había deleciones de una sola base en regiones homopolímeros de secuencia corta. Estas zonas fueron analizadas visualmente en IGV [44]. La figura 10 muestra un ejemplo de este tipo de eventos.

A

B

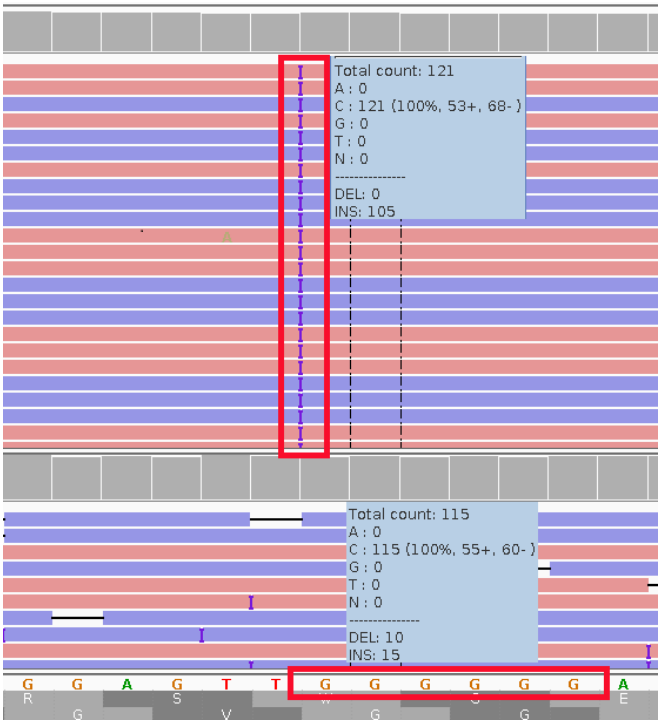

Figura 10 Visualización mediante IGV.
Representación de región que presenta homopolímeros de G (recuadro rojo horizontal) en el cromosoma 8 del nuevo ensamblaje de *L.major*. Panel (A): Visualización del alineamiento de lecturas de Illumina (arriba) y PacBio (abajo) frente al genoma ensamblado. Panel (B): Región correspondiente a la región de interés (recuadro negro de Panel A) ampliada. La presencia de una inserción (Recuadro rojo vertical) esta presente en 105 lecturas de Illumina frente a un total de 121 lecturas en esa posición (cuadro azul arriba). En PacBio solo 15 lecturas de un total de 115 en esa posición presentan la inserción (cuadro azul abajo).

Como puede observarse en la figura, el consenso del ensamblaje, basado en las lecturas de PacBio, es consistente con las secuencias de las lecturas de PacBio que alinean en esta región (solo 15 lecturas de un total de 115 contienen una inserción). Sin embargo, el alineamiento de las lecturas de Illumina apunta a la falta de una G en la posición marcada del genoma ensamblado. Más del 80% de las lecturas que caen en esa posición (105

lecturas con insercion con respecto a un total de 121 lecturas) presentan una inserción, la cual se ha comprobado que está presente en el genoma de referencia [8].

Por el momento, no hay demasiada información publicada sobre este tipo de problemas asociados a la tecnología de PacBio. Sin embargo, se ha reportado que a pesar de que dicha tecnología ha mejorado de forma espectacular, sigue habiendo errores sistemáticos en homopolímeros [45] sobre todo de nucleótidos de G o C [46]. Además, se ha visto que las inserciones de una sola base es lo más habitual, mientras que inserciones más largas son mucho más raras.

Estos datos sugieren que la tecnología PacBio podría tener dificultad para determinar la secuencia exacta en este tipo de regiones. Varios de estos indels caen en genes esenciales del parásito, de forma que la falta de una base altera el marco de lectura y en consecuencia la secuencia proteica predicha.

2.9 Corrección de secuencias del ensamblaje con lecturas de Illumina. Creación de un pipeline de corrección mediante el uso de distintas herramientas.

En base a los errores detectados, que se indican arriba, se llevó a cabo el diseño de un protocolo para la corrección del nuevo ensamblaje de *L. major* mediante el alineamiento sobre el mismo de lecturas de Illumina.

Herramientas bioinformáticas

- Pilon

Con la finalidad de mejorar el ensamblaje del genoma, se usó Pilon [47], una herramienta que, de forma automática, encuentra inconsistencias entre el ensamblaje y el alineamiento de lecturas frente a dicho ensamblaje en formato BAM [48]. Este proceso es muy útil ya que la tecnología de PacBio puede fallar en regiones donde hay homopolímeros o alternancias GC, por tanto, siendo *Leishmania* un genoma con alto contenido en G+C es muy recomendable llevar a cabo este proceso. El comando usado fue el siguiente:

```
$ java -jar Pilon.jar -genome <genome.fasta > <alignment.bam>
```

- PacBio-utilities

Dado que no todas las herramientas usan los mismos algoritmos, y estos no son perfectos, el ensamblaje corregido por Pilon se analizó mediante PacBio-utilities [49], una herramienta basada en la creación de scripts específicos para problemas observados en ensamblajes con lecturas de PacBio. Mediante el uso de archivos de alineamiento en formato BAM de lecturas Illumina frente a los ensamblajes de lecturas de PacBio, se indican en primer lugar posibles inserciones y deleciones en el genoma ensamblado, generando una secuencia genómica corregida.
Para evitar inconsistencia en los cambios, se estableció que la cobertura mínima para considerar una deleción o inserción debía ser de 10 lecturas, dado que es el filtro mínimo requerido para considerar variantes [50].
Por otro lado, se estableció que la fracción de lecturas que apoyan la presencia de una inserción o deleción en dicha posición fuese mayor o igual al 80%. Aquellos cambios por debajo de dicho filtro se consideran variantes polimórficas (polimorfismo alélico, *Leishmania* tiene un genoma diploide).

Como resultado, 143 posiciones fueron corregidas mediante esta primera aproximación. Los comandos usados fueron los siguientes:

$ pacbio-util indel-targets -f genoma.fasta alineamiento.bam > targets.txt

$pacbio-util indel-apply -f genoma.fasta -t targets.txt > genoma_corregido.fasta

- FreeBayes.

FreeBayes [51] es un detector de variantes bayesiano diseñado para encontrar pequeños polimorfismos e indels (deleciones o inserciones) entre otros, basándose en el alineamiento de lecturas frente a un ensamblaje.

En este caso, el ensamblaje corregido por PacBio-utilities se analizó mediante FreeBayes con los mismos parámetros. Un total de 42 indels nuevos (28 inserciones y 14 deleciones) fueron detectados mediante este programa.

Finalmente, los errores detectados con freebayes fueron corregidos de nuevo con el software PacBio-utilities que, a pesar de que también encontraba dichas variantes, no pasaban el filtro de calidad del programa, por tanto, se comprobó manualmente la calidad de dichas variantes confirmando que se trataba verdaderamente de errores de la secuenciación de PacBio y fueron corregidos.

Resultados

Tras finalizar este proceso, un total de 185 posiciones fueron corregidas mediante el uso de alineamientos con lecturas de Illumina. Teniendo en cuenta que previamente se usó Pilon que consiguió corregir 4005 indels, un total de 4190 inserciones y deleciones fueron erróneamente introducidas en el genoma ensamblado mediante la tecnología de PacBio. Aunque este problema no fue previsto y, por tanto, no incluido inicialmente en el plan de trabajo, una vez identificado, se consideró esencial llevar a cabo esta corrección, ya que la secuencia de algunos genes podría verse significativamente afectada por la inserción o deleción de una sola base. Posiblemente, muchos de los genes corregidos podrían haber sido anotados como pseudogenes o las secuencias aminoacídicas anotadas erroneamente.

Posteriormente, se llevó a cabo un análisis de la cobertura total de ambos ensamblajes con las lecturas de Illumina y PacBio que se alinearon frente al genoma con Bowtie2 [52] y BLASR [53] respectivamente. Los comandos usados se detallan a continuación.

```
$ bowtie2 --local -p 24 -k 1 -x db -q -1 read1.fastq -2 read2.fastq
-S alignment.sam
```

```
$ pbalign "archivo.fofn" "folder_reference" "aligned_reads.sam"—
seed=1 --minAccuracy=0.75 --minLength=50 -algorithmOptions= "-
useQuality"—algorithmOptions=' -minMatch 12 -bestn 10 -
minPctIdentity 70.0'—hitPolicy=randombest—tmpDir=tmpdir –nproc=24 -
-regionTable=filtered_regions.fofn
```

El análisis de cobertura se realizó utilizando la herramienta GenomeCoverageBed [54].

Los gráficos de cobertura se generaron con GNUPLOT [55]. En el anexo II se detalla el código creado para generar las figuras.

Los cromosomas fragmentados y posteriormente unidos mediante minimus2, fueron

comparados frente a la referencia en base a la cobertura. Las Figura 11 muestra el perfil de cobertura de los cromosomas 8, 19, 22, 27 y 35 respectivamente, que se unieron con minimus2, en comparación con la cobertura de dichos cromosomas en la referencia actual.

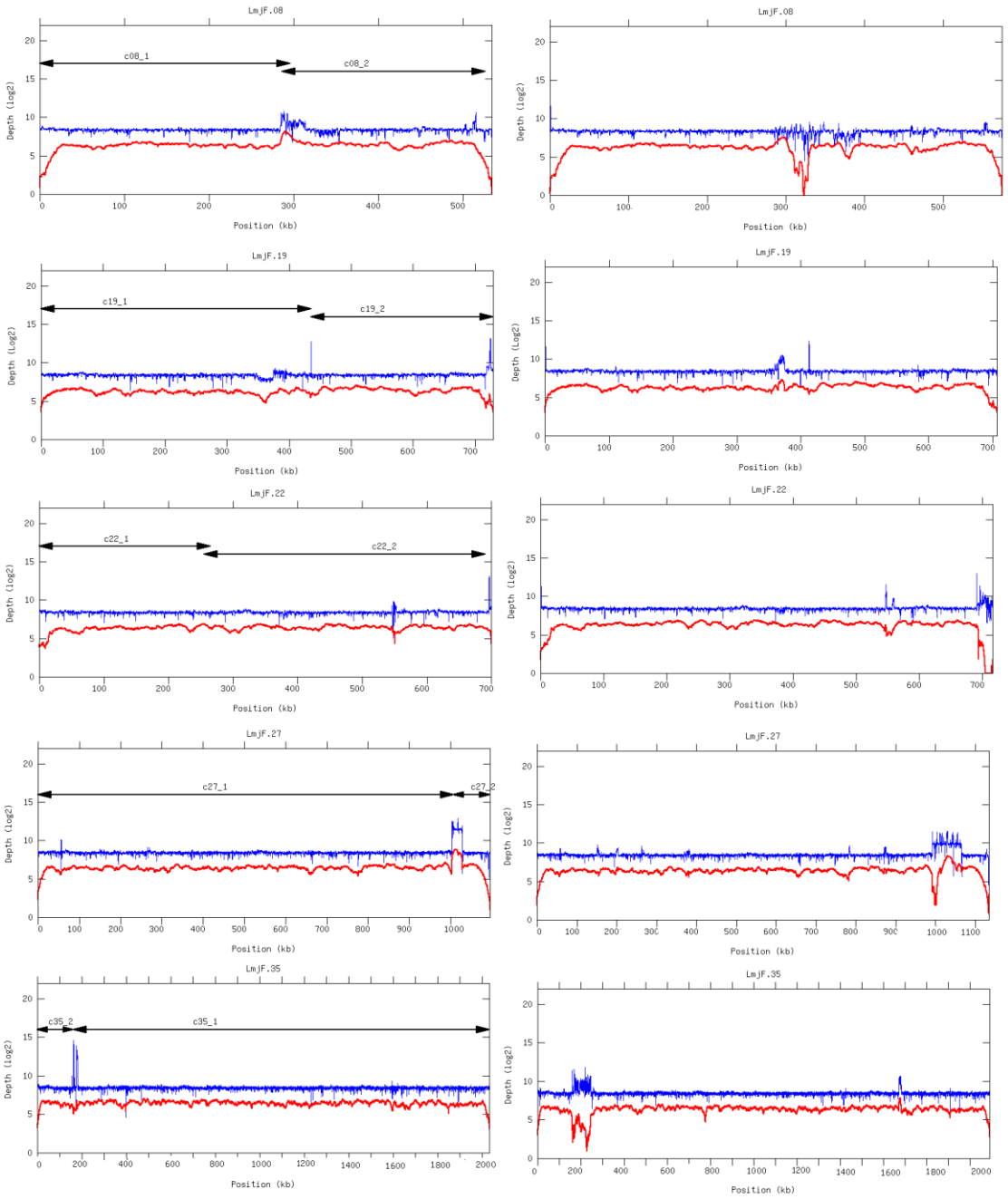

Figura 11 Análisis de cobertura de cromsomas fragmentados

Visualización de cobertura a lo largo de los cromosomas formados por la union de dos contigs ensamblados con PacBio (izquierda) y cromosomas de referencia (derecha). La cobertura se determinó mediante el análisis de ventanas deslizantes (bin 200 pb) con lecturas Illumina (en azul) y PacBio (en rojo), a lo largo de los cromosomas 8, 19, 22, 27 y 35. Los tamaños de los contigs se muestran por flechas. Todos los cromosomas se unieron con la herramienta minimus2.

Como puede observarse en la figura 11, las zonas de unión de los contigs resultantes del ensamblaje de PacBio, presentan una cobertura poco homogénea cuando se alinean las lecturas sobre los cromosomas del genoma de referencia, lo que pone de manifiesto la dificultad de ensamblaje y secuenciación de dichas regiones. A pesar de ello, se puede observar una mejora sustancial de los cromosomas ensamblados en este proyecto a nivel de cobertura de lecturas, indicando que el proceso de ensamblaje y unión de contigs han sido adecuados.

2.10 Identificación de grandes cambios respecto al genoma de referencia.

Identificación de regiones genómicas no catalogadas: Análisis de zonas de inserción y deleción en el genoma de referencia con respecto al actual, estudio de genes afectados en el genoma de referencia e identificación de nuevos genes en el nuevo ensamblaje.

Una vez llevado a cabo el proceso de corrección, se procedió a identificar posibles grandes diferencias genómicas entre el genoma de referencia (GeneDB.org) y el nuevo ensamblaje (este proyecto).

Como primera aproximación, se llevó a cabo un proceso de análisis basado en la realización de alineamientos múltiples con la herramienta MAFFT entre los cromosomas del genoma de referencia y el nuevo, de forma individual. Como resultado, se detectaron un número significativo de regiones afectadas. Sin embargo, tras un estudio detallado de dichas regiones, se constató que algunos de los resultados eran artefactos generados por el alineador, que mostró dificultad en alinear regiones con repeticiones génicas, una característica muy común de estos parásitos. Por ello, fue preciso buscar estrategias alternativas, para lo que se llevó a cabo un estudio de la bibliografía relacionada con esta temática.

Finalmente, se decidió abordar este análisis mediante la utilización de varias herramientas bioinformáticas y la realización de scripts en Python.

Herramientas bioinformáticas

- AsmVar

AsmVar [56] es un software para la detección, el genotipado y la caracterización de variantes estructurales y de nuevas secuencias, con resolución a nivel de nucleótido, desarrollado para el estudio de ensamblajes *de novo.*

AsmVar requiere un archivo de alineamiento entre ambas parejas de cromosomas que se llevó a cabo con LAST [40]. La principal diferencia entre este alineador y el usado anteriormente es que LAST está desarrollado para enfrentarse de manera más eficiente con secuencias ricas en repeticiones. Los comandos usados fueron:

```
$lastdb -cR11 -uNEAR l_major_ref reference.fasta

$lastal -m50 -E0.05 l_major_ref query.fasta | last-split -m1 > last.maf

$ ASV_VariantDetector -i <last.maf> -t reference.fasta -q query.fasta -s LmjF -o LmjF_asmvar
```

- Assemblytics

Para no restringir el estudio a un solo algoritmo, se planteó utilizar esta herramienta para validar y complementar los resultados obtenidos con la herramienta anterior. Assemblytics [57] es una herramienta desarrollada para detectar y analizar variantes de un ensamblaje *de* novo frente a un genoma de referencia; este programa incluye un filtro de anclaje para evitar errores causados por la presencia de secuencias repetitivas. Emplea el alineador nucmer [39] que detecta variantes a gran escala. Los comandos utilizados fueron:

```
$ nucmer –maxmatch –l 100 –c 500 reference.fasta query.fasta –
prefix LmjFNew

$Assemblytics LmjFNew.delta LmjFNew 70000
```

Una vez completados ambos procesos, se unificaron las variantes encontradas mediante ambos programas y se obtuvo un archivo en formato BED [58] con los cambios detectados entre cada pareja de cromosomas.

Con el objetivo de identificar los genes afectados por dichos cambios, se usó la herramienta intersectBed de bedtools [52], que, a partir de las coordenadas de un archivo de anotación del genoma de referencia, detecta si hay genes afectados o no en las regiones donde se han detectado cambios. El comando fue:

```
$ intersectBed variantes.bed AnotacionReferencia.bed >
GenesAfectados.bed
```

Finalmente, se decidió analizar aquellos genes afectados cuya inserción o deleción fuera mayor o igual a 100 bases. Por debajo de este umbral, se han detectado 343 cambios que afectan a genes, sin embargo, la mayoría corresponde a deleciones de entre 1 y 10 bases y un máximo de 40 bases delecionadas.

Mediante la realización de un script en Python se generaron archivos con las localizaciones de las regiones afectadas por inserciones o deleciones mayores o iguales a 100 bases, y la posterior identificación de los genes implicados. En el anexo III se muestra el script utilizado.

Resultados

Un total de 167 regiones con inserciones y deleciones de entre 100 bases y 170 kb de longitud fueron detectadas mediante el uso de estas herramientas. Dichas regiones, afectan a 175 genes. Al haberse seleccionado regiones con un mínimo de 100 bases de longitud se elimina la posibilidad de que estas diferencias sean debidas a razones biologicas dado que se trata de regiones muy amplias que afectan a secuencia codificante del parásito que pueden ser esenciales. En la tabla suplementaria S1 se incluye una lista de todos los genes afectados junto con su localización en el genoma de referencia.

2.11 Identificación de regiones ordenadas de forma errónea.

Uso de herramientas bioinformáticas para la visualización de aquellas regiones mal ensambladas en el genoma de referencia con respecto a la nueva versión de la secuencia del genoma.

Después del proceso de anotación, se procedió a analizar las 167 regiones (inserciones y deleciones) que fueron detectadas y clasificadas anteriormente. Cada zona fue analizada

junto con los diagramas de cobertura para cada cromosoma. En todos los casos, las deleciones o inserciones detectadas coinciden con grandes bajadas o incrementos de cobertura con respecto a la cobertura media, en los cromosomas del genoma de referencia de *L. major*. Sin embargo, la cobertura de dichas regiones en el nuevo ensamblaje muestra un perfil más homogéneo. Estos estudios permiten concluir que ha habido una mejora sustancial en la secuencia genómica de *L. major*; habiéndose determinado con mayor precisión el número total de genes y el número de copias de genes repetidos.

Es importante destacar que, en la mayoría de los casos, las regiones afectadas se localizan en zonas repetitivas del genoma lo que provoca que, en el caso de las deleciones, disminuya el número de copias de los genes afectados y en el caso de las inserciones, se incremente el número de copias. En otras ocasiones, no aumenta o disminuye el número de copias, sin embargo, la longitud de los genes afectados es mayor o menor con respecto a la referencia dependiendo de si está afectado por una inserción o una deleción.

En las figuras 12 y 13 se muestran dos ejemplos de regiones mal ensambladas en el genoma de referencia y cómo se han corregido en el nuevo genoma.

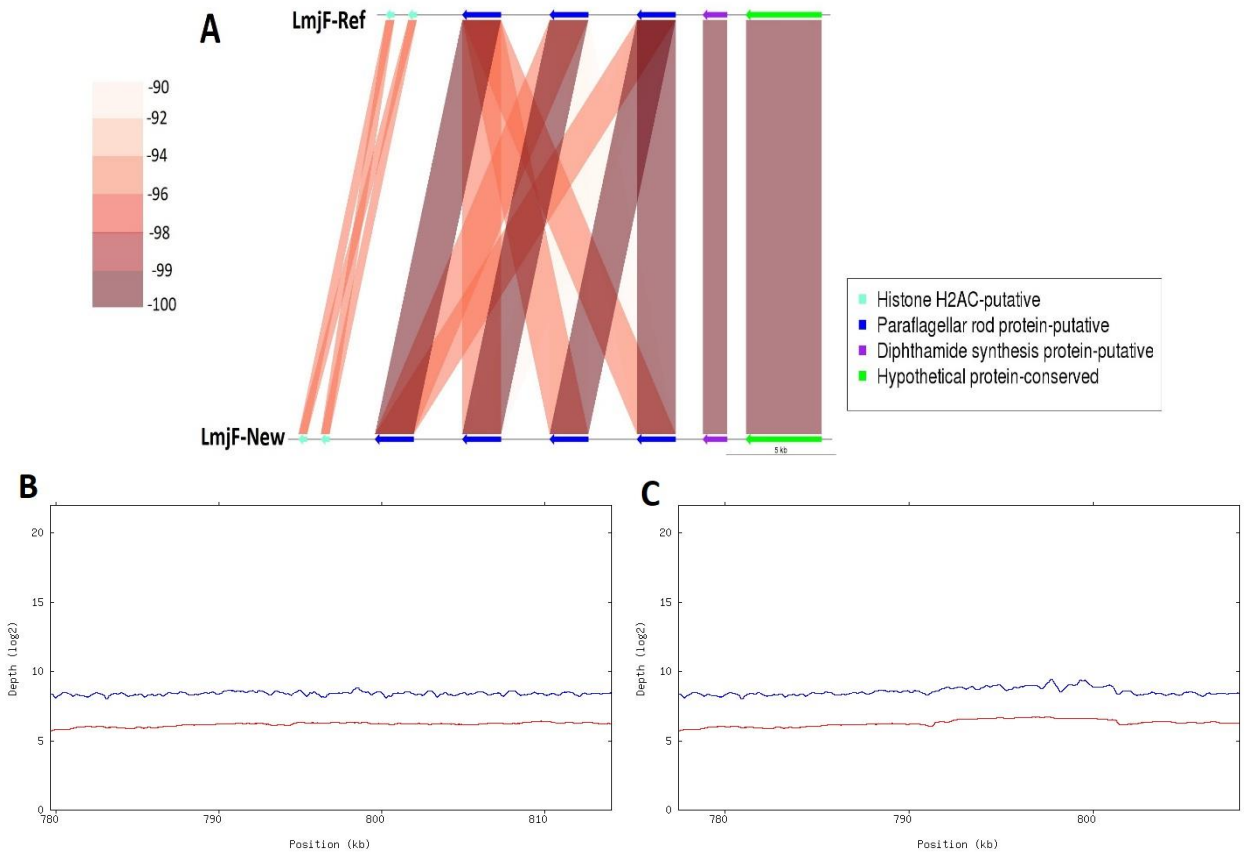

Figura 12. Número de copias génicas en el locus de proteína rod paraflagelar en el cromosoma 29.
Panel (A): Estructura genómica de la región que contiene el locus rod paraflagelar en el genoma de *L. major* Friedlin de referencia (LmjF-Ref) y en el ensamblaje realizado (LmjF-New). El porcentaje de identidad del alineamiento BLAST se muestra sombreando en marrón (véase la escala de color). Panel (B): Distribución de lecturas de Illumina (en azul) y de PacBio (en rojo) a lo largo de la región que contiene el locus génico de la proteína rod paraflagelar en el genoma ensamblado en este trabajo. Panel (C): Distribución de lecturas de Illumina (en azul) y de PacBio (en rojo) a lo largo de la región genómica estudiada usando como referencia el genoma previo de *L.major* [8].

La figura 12 muestra una región de genes repetidos en el cromosoma 29 de este parásito. Como puede observarse, tanto el gen de la histona H2AC como el gen que codifica para la proteína paraflagelar rod tienen dos y tres copias en tándem respectivamente, en el cromosoma de referencia. Sin embargo, aunque se mantiene el mismo número de copias para la histona H2AC en el nuevo ensamblaje, este último cuenta con una copia adicional del gen de la proteina paraflagelar rod. La distribución homogénea de la cobertura por parte de lecturas de PacBio y de Illumina apoya que el genoma ensamblado en este trabajo es más certero.

Por el contrario, también se han encontrado cambios que suponen una disminución en el número de copias. Así, como muestra la figura 13, existe una importante discrepancia en cuanto al número de genes HSP83/90 ('heat shock protein 83-17'), localizados en el cromosoma 33. El genoma de referencia muestra la existencia de 17 genes dispuestos en tándem, mientras que el nuevo genoma ensamblado sólo cuenta con 12 copias. Como en el caso anterior, dada la distribución homogénea de las lecturas de PacBio e Illumina a lo largo del nuevo genoma ensamblado (panel C), y la bajada de cobertura para lecturas de Illumina y PacBio en dicha región del genoma de referencia, estos datos sugieren una mayor exactitud en el número de copias ensamblados en el nuevo genoma de *L. major*.

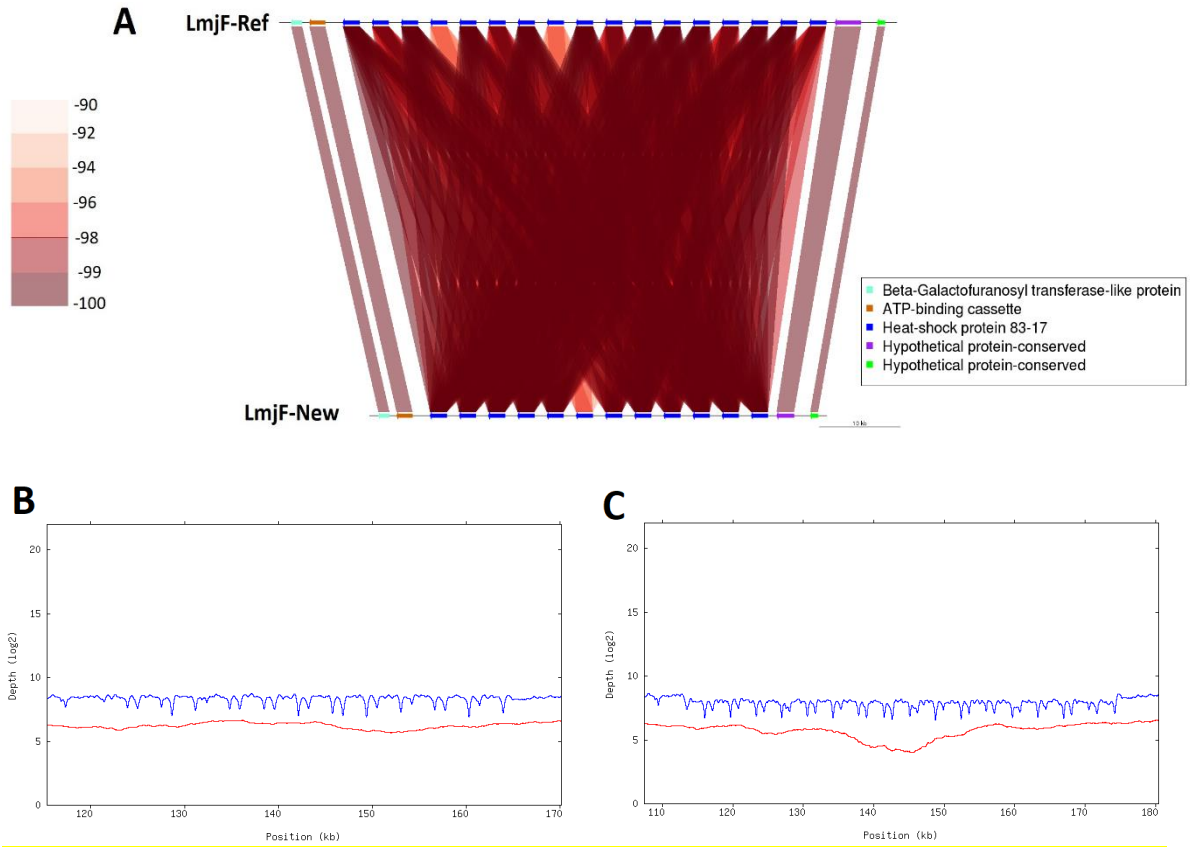

Figura 13. Número de copias génicas en el locus de hsp-83-17 en el cromosoma 33.
Panel (A): Estructura genómica de la región que contiene el locus hsp-83-17 en el genoma de *L. major Friedlin* de referencia (LmjF-Ref) y en el ensamblaje realizado en este proyecto (LmjF-New). El porcentaje de identidad del alineamiento BLAST se muestra sombreando en marrón (véase la escala de color). Panel (B): Distribución de lecturas de Illumina (en azul) y PacBio (en rojo) a lo largo de la región que contiene el locus hsp-83-17 en el genoma ensamblado en este trabajo. Panel (C): Distribución de lecturas de secuencia de Illumina (en azul) y PacBio (en rojo) a lo largo del locus hsp-83-17 usando como referencia el genoma de *L. major* de referencia [8].

Cabe destacar que si bien se ha mejorado sustancialmente el genoma de *L. major*, algunas regiones repetitivas muy amplias pudieran no haber sido ensambladas de forma definitiva. Así, se ha detectado la existencia en algunos puntos de los cromosomas ensamblados la existencia de una cobertura de lecturas PacBio e Illumina que no es completamente homogénea; si bien, en todos los casos, la cobertura es sustancialmente más homogénea que cuando se alinean frente al genoma de referencia. En las figuras 14 y 15 se muestran ejemplos que ilustran estos hechos.

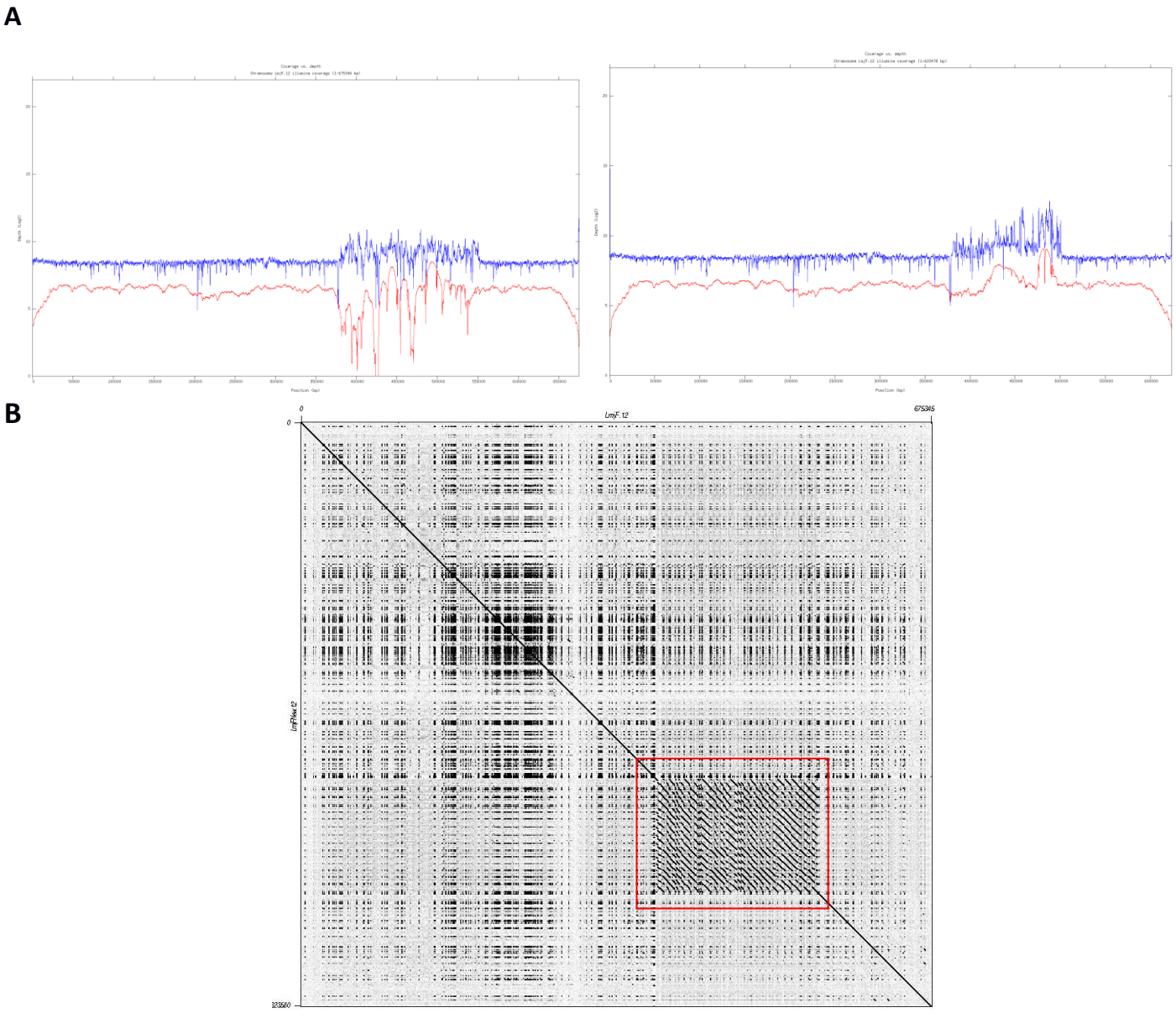

Figura 14. Análisis de cobertura a lo largo de dos cromosomas con grandes regiones repetitivas.
En los diagramas de puntos, las regiones idénticas tendrán una línea diagonal en el centro de la matriz. Las inserciones y deleciones entre secuencias dan lugar a interrupciones en esta diagonal y las regiones repetitivas dan lugar a coincidencias diagonales adicionales además de la diagonal central. Panel (A): Perfil de cobertura de lecturas sobre el cromosoma 12 de referencia (izquierda) o el ensamblado en este proyecto (derecha). Panel (B). Dotplot. Alineamiento del del cromosoma 12 de la referencia [5] (eje horizontal) frente al cromosoma 12 del genoma ensamblado (eje vertical) con Gepard.La region repetitiva se marca en rojo.

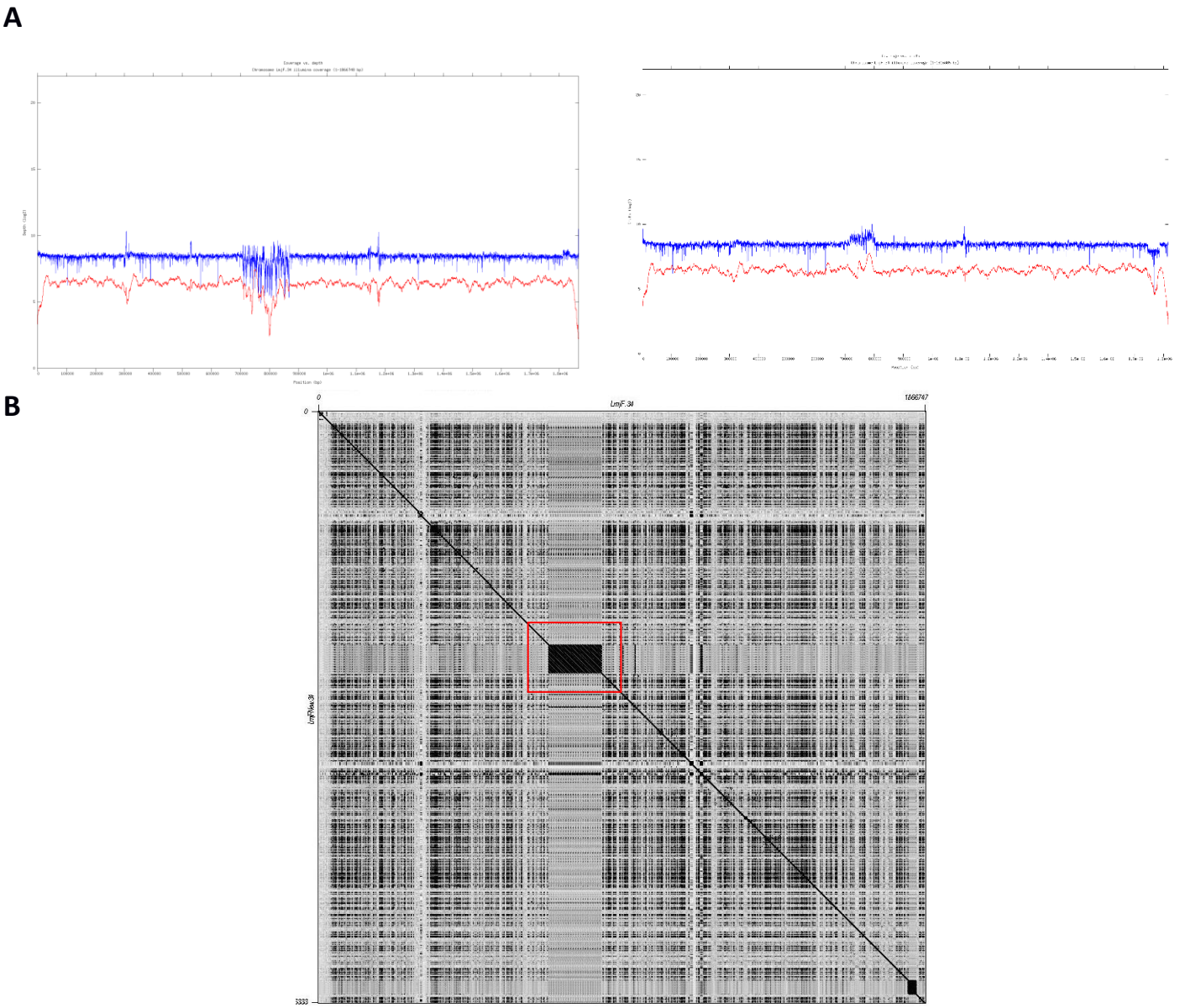

Figura 15. Análisis de cobertura a lo largo de dos cromosomas con grandes regiones repetitivas

En los diagramas de puntos, las regiones idénticas tendrán una línea diagonal en el centro de la matriz. Las inserciones y deleciones entre secuencias dan lugar a interrupciones en esta diagonal y las regiones repetitivas dan lugar a coincidencias diagonales adicionales además de la diagonal central. Panel (A): Perfil de cobertura de lecturas sobre el cromosoma 34 de referencia (izquierda) o el ensamblado en este proyecto (derecha). La cobertura se determinó mediante el análisis de ventanas deslizantes (bin 200 pb) con lecturas Illumina (en azul) o PacBio (en rojo). Panel (B). Dotplot. Alineamiento del del cromosoma 34 de la referencia [5] (eje horizontal) frente al cromosoma 12 del genoma ensamblado (eje vertical) con Gepard.La region repetitiva se marca en rojo.

Las regiones donde se pierde la homogeneidad de la cobertura en las figuras 14 y 15 en los cromosomas de referencia, (cromosoma 12 y 34 respectivamente) coinciden con grandes regiones de repetición génica (entre 10 y 15 kb de repetición). Ambas regiones analizadas presentan una longitud mayor en el genoma de referencia en comparación con el nuevo ensamblaje que como puede observarse en el dotplot generado mediante Gepard [37].

Posteriormente, se procedió a la identificación de regiones ordenadas de forma errónea. La comparación entre el genoma de referencia y el nuevo ensamblaje ha permitido identificar reorganizaciones en algunos cromosomas. En la figura 16, se puede observar la existencia de una región invertida de alrededor de 7.5 kb de longitud (amarillo) en el genoma de referencia con respecto al nuevo ensamblaje. Como parte del análisis, se incluyó el cromosoma 29 del genoma de *L.infantum* [10] que a pesar de ser otra especie

de *Leishmania*, comparte alrededor del 98% de identidad de secuencia génica con *L. major* [59] Tal y como se observa, esa región está ensamblada del mismo modo en *L. infantum* y en el nuevo ensamblaje de *L. major.*

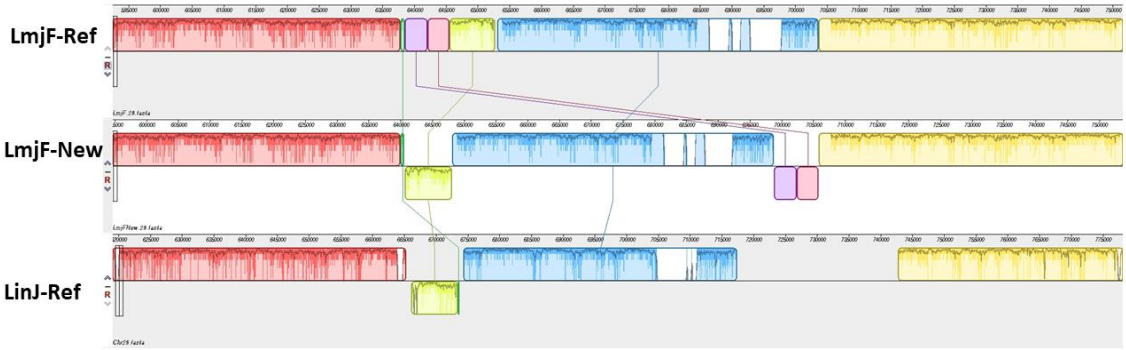

Figura 16 Esquema de regiones mal ensambladas en el genoma actual de *L.major*.
Los bloques de sintenia están representados en diferentes colores en el cromosoma 29 (Panel A) tras la comparación por pares entre el genoma de *L.major* de referencia (LmjF-Ref), el genoma recién ensamblado (LmjF-New) y el genoma de referencia de *L. infantum* (LinJ-Ref). Los alineamientos por pares fueron realizados con la herramienta MAUVE, que usa códigos de color para representar bloques de regiones conservadas. Las secciones ubicadas debajo del eje-X muestran eventos de inversión.

Además, con la finalidad de confirmar la calidad del ensamblaje en esta región, se estudió el cromosoma a nivel de cobertura. En la figura 17, se muestra el perfil de cobertura para el cromosoma 29 de referencia y el cromosoma ensamblado en este proyecto. La homogeneidad de la cobertura para ambos tipos de lecturas en el nuevo cromosoma en comparación con el perfil irregular en la región de interés (en torno a las 650 kb), permite concluir que dicha región está ahora correctamente ensamblada.

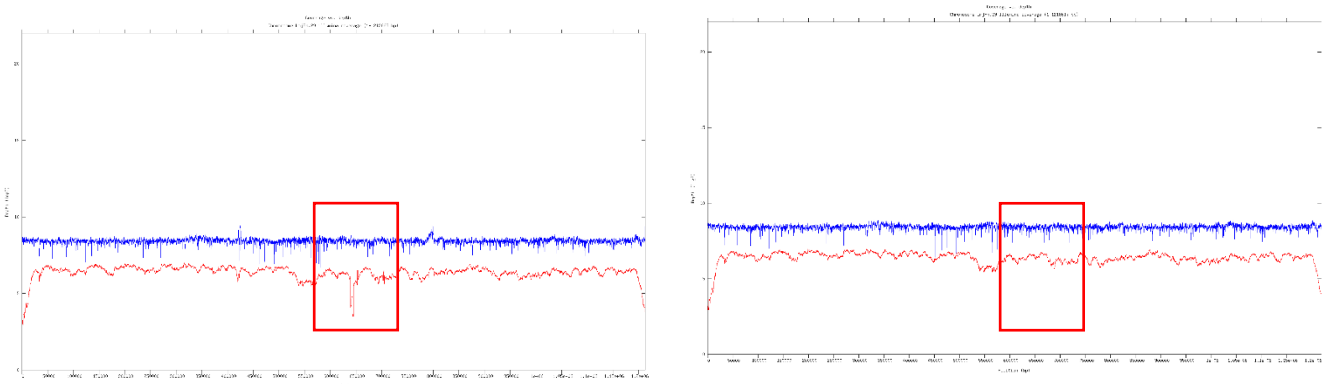

Figura 17 Análisis de cobertura a lo largo del cromosoma 29.
Cobertura del análisis del genoma de referencia (izquierda) y del nuevo ensamblaje (derecha). La cobertura se determinó mediante el análisis de ventanas deslizantes (bin 200 pb) con lecturas Illumina (en azul) o PacBio (en rojo). La región mal ensamblada en el genoma de referencia con respecto al nuevo genoma está recuadrada en rojo.

Además de lo expuesto anteriormente, se propuso verificar la presencia en el nuevo ensamblaje de las 7 regiones genómicas añadidas a la primera versión del genoma de *L. major* (actualmente existente en la base de datos GeneDB) tras el trabajo de Alonso y colaboradores [8], que identificaron colapsos en el ensamblaje ocasionados por la presencia de secuencias repetitivas próximas. Con el objetivo de corroborar la calidad del ensamblaje realizado por estos autores, se analizó la presencia de los nuevos genes identificados (LmjF.15.1475, LmjF.15.0285, LmjF.24.0765, LmjF.14.0860, LmjF.19.0305, and LmjF.27.2035, LmjF.15.1480 y LmjF.27.2030) mediante una búsqueda con la herramienta BLAST sobre el genoma ensamblado en este trabajo. Los resultados mostraron que todos los genes se encuentran presentes en el nuevo genoma, lo que es un apoyo adicional a la solidez del genoma ensamblado en este trabajo.

3. Anotación del genoma ensamblado.

Uno de los principales procesos en la realización de un ensamblaje *de novo* es la anotación funcional y estructural de dicho genoma.

La anotación del nuevo ensamblaje de *L. major* se realizó utilizando el servidor web Companion [60] tomando la versión 6 del genoma de *L. major* de referencia (GeneDB.org) usando todos los parámetros por defecto y proporcionando únicamente la secuencia del nuevo ensamblaje.

Companion, es un anotador desarrollado específicamente para la anotación de genomas de tripanosomátidos, que incorpora varias herramientas de anotación. En primer lugar, lleva a cabo una anotación estructural con técnicas basadas en homología y *ab initio*. El anotador RATT [61], se usa para transferir modelos de genes altamente conservados con poca o ninguna diferencia con la referencia seleccionada. Por otro lado, Companion usa SNAP [62] y AUGUSTUS [63] como métodos de predicción *ab initio*.
Los pseudogenes se anotan usando alineamientos de proteína-ADN, que son generados mediante el alineador LAST [40].
En el caso de los ARN no codificantes, Companion usa el anotador *ab initio* ARAGORN [64] para la anotación de ARNs de transferencia, e INFERNAL [65] para otros tipos de ARN no codificantes.
Finalmente, se lleva a cabo la anotación funcional de cada gen codificante (descripción del producto proteico, nombre de genes, términos GO, identificador de ortólogos y parálogos) que se transfiere de anotaciones asociadas con genes de referencia determinados por OrthoMCL [66], una herramienta diseñada para la identificación de genes ortólogos o anotación de dominios Pfam-A [67], si no se pueden determinar ortólogos.

A pesar de que Companion toma la anotación del genoma de *L. major* como referencia, algunos de los genes encontrados en el nuevo ensamblaje (sobre todo los genes detectados por anotadores *ab initio*) carecen de correspondencia con genes de *L. major*, por no estar anotados en las bases de datos de este genoma.

Dada la importancia de mantener los nombres actuales de identificación (ID) de genes de *L. major*, y con la finalidad de analizar en detalle aquellos genes anotados por Companion, pero carentes de una correspondencia a genes anotados en el genoma de referencia de *L. major* [8], se analizaron las secuencias de proteinas proporcionadas por Companion mediante búsquedas con OrthoMCL, que en algunos casos encontró correspondencia

tanto con genes de *L. major* y como con genes de otras especies de *Leishmania*. La secuencia proteica de éstos últimos, se alinearon frente a los transcritos de la referencia de *L. major* [8] mediante tBLASTn. El análisis de las proteínas para las que no se identificaron ortólogos con OrthoMCL, se realizó mediante búsqueda BLASTp sobre una base de datos con secuencias no redundantes (nr), generada a través de las traducción de los genes depositados en GenBank junto con secuencias de otros bancos de datos (Refseq, PDB, SwissProt, PIR y PRF).

Finalmente, todos los datos generados por los tres tipos de análisis se combinaron en un archivo GFF3 [68] mediante un script en Python. En el Anexo IV se muestran las instrucciones contenidas en este script. La figura 18 muestra el flujo de trabajo llevado a cabo en este punto.

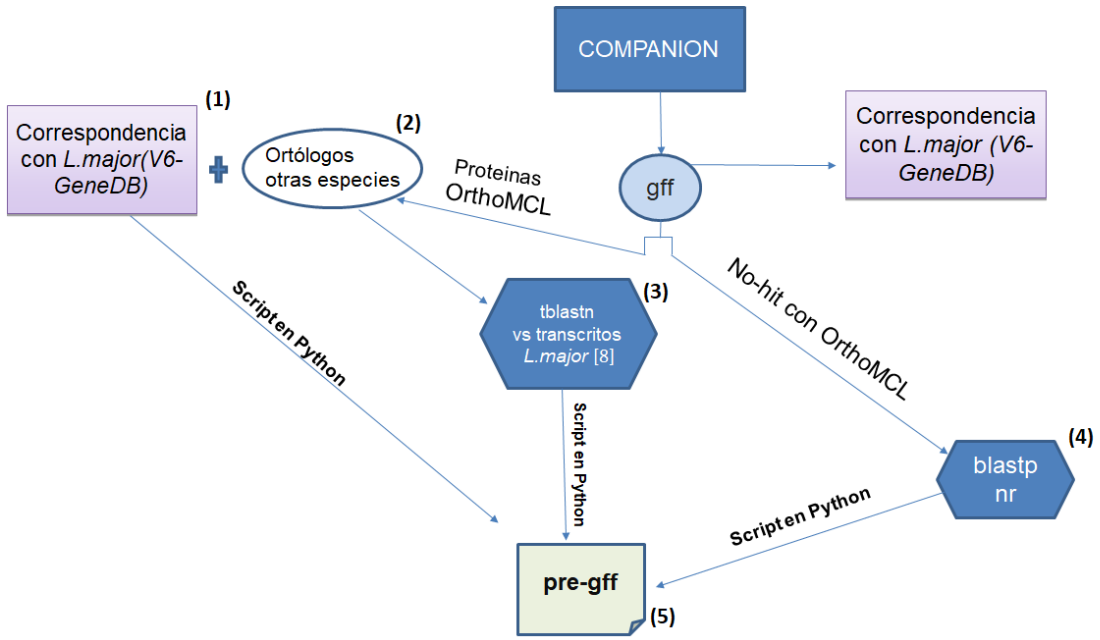

Figura 18 Flujo de trabajo llevado a cabo tras la anotación con el software Companion.
El archivo de anotación (gff) proporcionado por Companion carecía de algunas correspondencias con genes del genoma de referencia de *L.major*. Las secuencias de proteinas obtenidas fueron analizadas por OrthoMCL que encontró correspondencia con *L. major* **(1)** pero también con otras especies de *Leishmania* **(2)**. Las secuencias proteicas de éstos últimos genes fueron enfrentadas a los transcritos de referencia de *L. major* [5] mediante tBLASTn **(3)**. Aquellas proteínas sin resultado en OrthoMCL fueron analizadas mediante BLASTp nr **(4)**. Toda la información, fue incorporada al archivo final de anotación **(5)** mediante la generación scripts en Python.

4. Envío de secuencias a la base de datos de ENA.

El objetivo de este punto es describir el proceso seguido para depositar los datos de lecturas NGS en el European Nucleotide Archive (ENA). ENA es un repositorio europeo y de acceso libre que proporciona un registro completo de la información de proyectos de secuenciación de ADN y ARN llevados a cabo en todo el mundo. Permite acceder a datos crudos de secuenciación y obtener información complementaria como procedimientos experimentales, ensamblaje y anotación de secuencias y otros metadatos relacionados con los proyectos [69].

Para llevar a cabo el envio de las lecturas de Illumina, se deben comprimir los dos archivos fastq (correspondientes a las lecturas forware y reverse). Posteriormente, para cada uno de estos archivos comprimidos, se deben obtener sus números MD5.

Para el caso de las lecturas crudas de PacBio (en formato bax.h5) y los metadatos (metadata.xml), debe crearse un llamado 'manifest.all' que contiene la lista de todos las lecturas y metadatos que se transfieren a ENA. Los números MD5 también deben generarse para todos los archivos subidos.

Una vez obtenidos todos los archivos necesarios en su formato correspondiente se usó Filezilla[70] que permite transferir archivos, en este caso, a la red de ENA.

Después de transferir los datos, se creó el estudio del proyecto(que contendrá un resumen del tipo de estudio y análisis llevado a cabo.) mediante el servidor web de ENA. Por otro lado, también a través del servidor web, se enviaron las secuencias crudas de Illumina y PacBio cargadas previamente mediante Filezilla.

Los archivos del proyecto se mantendrán en privado hasta el 8/04/2020. El tiempo máximo de confidencialidad de los datos es de 2 años renovables. En el Anexo V se muestran los acuses de recibo de ENA que corroboran el correcto envio de los datos.

El números de acceso de ENA al estudio fue: PRJEB25921

5. Conclusiones

Disponer de un genoma robusto, sin fisuras y exento de colapsos es esencial para estudios que abarcan los campos de la genómica, transcriptómica o proteómica. La secuenciación de genomas ha ido avanzando de forma exponencial a lo largo de los años, lo que ha permitido mejorar la calidad de los trabajos relacionados con la secuenciación de genomas. Sin embargo, por el momento, no existe una tecnología perfecta que permita ensamblar adecuadamente genomas complejos como es el caso de *Leishmania*.

El presente estudio ha servido para comparar varias metodologías de secuenciación y ensamblaje concluyendo que a pesar de que existen numerosas herramientas para ensamblar lecturas de Illumina, donde el ensamblador SPAdes mostró los mejores resultados, los ensamblajes obtenidos estaban muy fragmentados y mostraban un número importante de regiones colapsadas.
A diferencia de lo que se pensaba al inicio del estudio, los ensamblajes híbridos no han mostrado una mejora sustancial con respecto al ensamblaje llevado a cabo únicamente con lecturas de Illumina y fueron desechados. Los ensamblajes no-híbridos llevados a cabo por HGAP con lecturas de PacBio, mostraron los mejores resultados de ensamblaje, ya que ofrece contigs más largos y menos fragmentados que los anteriores.

En este trabajo se ha demostrado que las dificultades para el completo ensamblaje de cromosomas a partir de lecturas de PacBio aparecen en zonas altamente repetitivas con longitudes de entre 20 y 60 kb, donde las lecturas de PacBio también colapsan durante el ensamblaje. El conocimiento obtenido y la utilización de herramientas bioinformáticas han permitido unir de forma precisa cromosomas fragmentados en varios contigs, y alargar la secuencia de los extremos de los cromosomas, para lo que las lecturas de Illumina han resultado muy útiles.

Además, se han detectado ciertas zonas del genoma cuya secuenciación mediante la tecnología de PacBio puede dar lugar a Indels debido a las características de la secuencia de ADN en este organismo y por la presencia de secuencias de homopolímeros, sobre todo de G y C, que se aparecen distribuídas por todo el genoma. Este hecho, ha complicado el desarrollo de del trabajo propuesto, pues no se anticipó, dada la alta cobertura media de lecturas de PacBio que disponíamos, que la secuencia consenso del ensamblaje con lecturas de PacBio no fuera definitiva.
La combinación de varias herramientas de corrección de genomas mediante secuencias de Illumina ha resultado esencial en este punto. Dada la novedad y la falta de información reportada, relacionada con este problema técnico, es necesario alertar a la comunidad científica sobre estos errores de secuencición y desarrollar filtros adecuados para corregirlos, como los propuestos en este trabajo.

El proceso de ensamblaje, unión, extensión y corrección pone de manifiesto la importancia de combinar ambos tipos de tecnologías (PacBio e Illumina) en los proyectos de ensamblaje de novo de genomas, como el que aquí se presenta.

Por otro lado, la existencia de un genoma de referencia de *L. major* bastante bien anotado y corregido [7] [8] ha permitido la realización de una precisa comparación con la finalidad de localizar grandes diferencias entre ambos genomas, que, en su mayoría, corresponden

a regiones repetitivas mal ensambladas en la referencia y con una cobertura más homogénea en el nuevo genoma ensamblado. En total se han detectado 175 genes afectados en el genoma de referencia que cuentan con más o menos copias en el nuevo ensamblaje dependiendo de si caen en regiones de inserción o deleción respectivamente. Este hecho supone una gran ventaja a todos los niveles puesto que ha permitido corregir el número de copias de varios genes de *L. major* de forma fiable. Sin embargo, a pesar de la clara mejora del genoma que ha supuesto este trabajo, se han detectado algunas regiones altamente repetitivas que posiblemente no se han ensamblado de forma definitiva.
La gran longitud de dichas repeticiones génicas supone un hándicap a la hora de llevar a cabo tanto el proceso de secuenciación como de ensamblaje de genomas. La generación de nuevos sistemas de secuenciación y/o algoritmos de ensamblaje en el futuro, podría ayudar a la corrección de la secuencia en estas complicadas regiones.

La generación de un flujo de trabajo como el realizado en el proceso de anotación, es esencial para planificar y llevar a cabo la curación completa del archivo de anotación que se enviará a las bases de datos públicas. El repositorio europeo ENA, permite el envío tanto de lecturas de secuenciación como de la anotación del genoma. El proceso de envío de datos permite, no solo almancenar de forma segura toda la información obtenida, sino también impulsar y dar a conocer los avances realizados al resto de científicos (y a la sociedad en general).

Es importante destacar que no sólo es necesario el uso de varias tecnologías de secuenciación, sino que también es importante un estudio, y el correcto uso, de las herramientas bioinformáticas especializadas actualmente existentes. La combinación de estos elementos ha permitido la obtención de un genoma más completo, y con mejoras sustanciales en la anotación, con respecto al genoma de referencia de este parásito actualmente existente en las bases de datos.

Por otro lado, este trabajo puede resultar de utilidad a la comunidad científica como guía al proporcionar un flujo de trabajo destinado al ensamblaje *de novo* de genomas a partir de la combinación de metodologías NGS.

Líneas de trabajo futuras

A pesar de haber alcanzado la mayoría de los objetivos inicialmente planteados, los errores intrínsecos a las técnicas NGS, y no anticipados, han impedido completar la revisión de la anotación informática realizada. Esta es una tarea a realizar previo al depósito de esta información en la base de datos de ENA.
Por otro lado, como trabajo futuro adicional se podrían desarrollar nuevas herramientas bioinformáticas para mejorar el ensamblaje de las zonas con un alto contenido en repeticiones.

Finalmente, El descubrimiento de errores de secuenciación en las zonas ricas en homopolímeros con la tecnología de PacBio, abre una puerta a la investigación del motivo por el cual ocurre ese acontecimiento para poder alertar a la comunidad científica de los posibles errores de secuencia que pueden tener los genomas secuenciados mediante este método y la búsqueda de protocolos de secuenciación que ayuden a minimizar dichos errores de secuenciación.

6. Glosario

- ***Ab initio:*** Procedimiento que usa solamente las propiedades de la secuencia de ADN para predecir localización de genes basándose en algoritmos que discriminan las regiones codificantes y no codificantes para inferir la localización de los genes.
- **Aneuploidía:** variaciones en el número de cromosomas.
- **Anotación estructural:** Encargada de detectar la posición (coordenades) de regiones codificantes de genes, la estructura de los exones e intrones (si procede) y predecir las secuencias de proteínas [71].
- **Anotación funcional:** Procedimiento mediante el cual se predice la función biológica de los genes y proteínas y su proceso biológico [71].
- **Cobertura:** Número de veces que un nucleótido es leído al considerar un conjunto de lecturas.
- **Contig:** Conjunto de lecturas que se han podido ordenar para formar un segmento continuo de secuencia en base al solapamiento de sus extremos [72].
- **Gap:** Segmento de secuencia sin determinar. Normalmente, se indican introduciendo un número de "N", que puede ser o no correcto.
- **G-cuádruplex:** Estructuras de ADN y ARN de orden superior constituidas por secuencias ricas en Guanina. Se forman alrededor de un núcleo de al menos dos tétradas apiladas de bases Guanina unidas por Puentes de Hidrógeno. Pueden formarse a partir de una, dos o cuatro hebras independientes de ADN (o ARN) y pueden adoptar una gran variedad de configuraciones según la dirección, la longitud y la secuencia de las hebras [73].
- **Homopolímero:** Son las repeticiones de secuencia simple más sencillas, poli (dA), poli (dT), poli (dG) y poli (dC). Están presentes en todos los genomas, pero en algunos eucariotas se encuentran en altas frecuencias [74].
- **Indel:** Contracción de las palabras inserción y deleción, y que invoca cualquiera de los dos eventos.
- **K-mer:** Todas las posibles subsecuencias (de longitud k) a partir de una lectura obtenida mediante secuenciación [75].
- **Lecturas paired-end** (lecturas pareadas). Proporciona la secuencia de ambos extremos de cada fragmento de ADN, siempre separados por una distancia aproximada conocida.
- **MD5:** Algoritmo de reducción criptográfico ampliamente usado. Uno de sus usos es el de comprobar que algún archivo no haya sido modificado tras su creación [76].
- **N50:** Longitud mínima del contig que se necesita para cubrir el 50% de la longitud total del genoma
- **Ortólogo:** Genes equivalentes u homólogos presentes en dos especies distintas.
- **Reordenamiento:** Los reordenamientos genómicos describen cambios mutacionales en el genoma tales como duplicación, deleción, inserción, inversión y translocación que son diferentes a un genoma de referencia. Por lo general el término solo se utiliza para describir cambios de ADN que van desde miles hasta a veces millones de pares de bases [77].
- **Scaffold:** Conjunto de contigs ordenados y orientados en base a la información obtenida de lecturas pareadas. A diferencia de los contigs, contiene huecos (gaps) de secuencia no determinada.

7. Bibliografía

[1] Smith, M., Bringaud, F., & Papadopoulou, B. (2009). Organization and evolution of two SIDER retroposon subfamilies and their impact on the Leishmania genome. BMC Genomics, 10(1), 240. http://dx.doi.org/10.1186/1471-2164-10-240

[2] Requena, J., Rastrojo, A., Garde, E., López, M., Thomas, M., & Aguado, B. (2017). Genomic cartography and proposal of nomenclature for the repeated, interspersed elements of the Leishmania major SIDER2 family and identification of SIDER2-containing transcripts. Molecular And Biochemical Parasitology, 212, 9-15. http://dx.doi.org/10.1016/j.molbiopara.2016.12.009

[3] Myler P, Fasel N (ed.) (2008) Leishmania: After The Genome. Caister Academic Press, Norfolk

[4] Requena JM (2011) Lights and shadows on gene organization and regulation of gene expression in Leishmania. Front Biosci 17:2069–2085. doi: 10.2741/3840

[5] Dumetz, F., Imamura, H., Sanders, M., Seblova, V., Myskova, J., Pescher, P., … Domagalska, M. A. (2017). Modulation of Aneuploidy in *Leishmania donovani* during Adaptation to Different *In Vitro* and *In Vivo* Environments and Its Impact on Gene Expression. *mBio*, *8*(3), e00599–17. http://doi.org/10.1128/mBio.00599-17

[6] Ivens, A. C., Peacock, C. S., Worthey, E. A., Murphy, L., Aggarwal, G., Berriman, M., … Myler, P. J. (2005). The Genome of the Kinetoplastid Parasite, Leishmania major. Science (New York, N.Y.), 309(5733), 436–442. http://doi.org/10.1126/science.1112680

[7] Rastrojo, A., Carrasco-Ramiro, F., Martín, D., Crespillo, A., Reguera, R. M., Aguado, B., & Requena, J. M. (2013). The transcriptome of Leishmania major in the axenic promastigote stage: transcript annotation and relative expression levels by RNA-seq. BMC Genomics, 14, 223. http://doi.org/10.1186/1471-2164-14-223

[8] Alonso, G., Rastrojo, A., López-Pérez, S., Requena, J., & Aguado, B. (2016). Resequencing and assembly of seven complex loci to improve the Leishmania major (Friedlin strain) reference genome. Parasites & Vectors, 9(1), 74. http://dx.doi.org/10.1186/s13071-016-1329-4

[9] Goodwin, S., McPherson, J., & McCombie, W. (2016). Coming of age: ten years of next-generation sequencing technologies. Nature Reviews Genetics, 17(6), 333-351. http://dx.doi.org/10.1038/nrg.2016.49

[10] González-de la Fuente, S., Peiró-Pastor, R., Rastrojo, A., Moreno, J., Carrasco-Ramiro, F., Requena, J., & Aguado, B. (2017). Resequencing of the Leishmania infantum (strain JPCM5) genome and de novo assembly into 36 contigs. Scientific Reports, 7(1), 18050. http://dx.doi.org/10.1038/s41598-017-18374-y

[11] Minoche, A., Dohm, J., & Himmelbauer, H. (2011). Evaluation of genomic high-throughput sequencing data generated on Illumina HiSeq and Genome Analyzer systems. Genome Biology, 12(11), R112. http://dx.doi.org/10.1186/gb-2011-12-11-r112

[12] Treangen, T., & Salzberg, S. (2011). Repetitive DNA and next-generation sequencing: computational challenges and solutions. Nature Reviews Genetics, 13(1), 36-46. http://dx.doi.org/10.1038/nrg3117

[13] Liao, Y.-C., Lin, S.-H., & Lin, H.-H. (2015). Completing bacterial genome assemblies: strategy and performance comparisons. *Scientific Reports*, *5*, 8747. http://doi.org/10.1038/srep08747

[14] Jayakumar, V., & Sakakibara, Y. (2017). Comprehensive evaluation of non-hybrid genome assembly tools for third-generation PacBio long-read sequence data. *Briefings in Bioinformatics*, Nov 3, 1–11 http://doi.org/10.1093/bib/bbx147

[15] Heydari, M., Miclotte, G., Demeester, P., Van de Peer, Y., & Fostier, J. (2017). Evaluation of the impact of Illumina error correction tools on de novo genome assembly. *BMC Bioinformatics*, *18*, 374. http://doi.org/10.1186/s12859-017-1784-8

[16] https://www.pacb.com

[17] www.sequencing.uio.no

[18] https://www.qiagenbioinformatics.com/products/clc-genomics-workbench

[19] https://www.qiagen.com/us/

[20] Bankevich, A., Nurk, S., Antipov, D., Gurevich, A. A., Dvorkin, M., Kulikov, A. S., … Pevzner, P. A. (2012). SPAdes: A New Genome Assembly Algorithm and Its Applications to Single-Cell Sequencing. Journal of Computational Biology, 19(5), 455–477. http://doi.org/10.1089/cmb.2012.0021

[21] Medvedev, P., Pham, S., Chaisson, M., Tesler, G., & Pevzner, P. (2011). Paired de Bruijn Graphs: A Novel Approach for Incorporating Mate Pair Information into Genome Assemblers. Journal of Computational Biology, 18(11), 1625–1634. http://doi.org/10.1089/cmb.2011.0151

[22] Liu, B., Liu, C.-M., Li, D., Li, Y., Ting, H.-F., Yiu, S.-M., … Lam, T.-W. (2016). BASE: a practical de novo assembler for large genomes using long NGS reads. BMC Genomics, 17(Suppl 5), 499. http://doi.org/10.1186/s12864-016-2829-5

[23] Al-okaily Anas A. (2016). HGAP: denovo genome assembly method for bacterial genomes using high coverage short sequencing reads. BMC Genomics, 17, 193. http://doi.org/10.1186/s12864-016-2515-7

[24] http://kmergenie.bx.psu.edu/

[25] Chin, C., Alexander, D. H., Marks, P., Klammer, A. A., Drake, J., Heiner, C., . . . Korlach, J. (2013). Nonhybrid, finished microbial genome assemblies from long-read SMRT sequencing data. Nature Methods, 10(6), 563-569. doi:10.1038/nmeth.2474

[26] Commins, J., Toft, C., & Fares, M. A. (2009). Computational Biology Methods and Their Application to the Comparative Genomics of Endocellular Symbiotic Bacteria of Insects. Biological Procedures Online, 11, 52–78. http://doi.org/10.1007/s12575-009-9004-1

[27] Miller, J. R., Zhou, P., Mudge, J., Gurtowski, J., Lee, H., Ramaraj, T., … Silverstein, K. A. T. (2017). Hybrid assembly with long and short reads improves discovery of gene family expansions. BMC Genomics, 18, 541. http://doi.org/10.1186/s12864-017-3927-8

[28] Koren, S., Schatz, M. C., Walenz, B. P., Martin, J., Howard, J., Ganapathy, G., … Phillippy, A. M. (2012). Hybrid error correction and de novo assembly of single-molecule sequencing reads. Nature Biotechnology, 30(7), 693–700. http://doi.org/10.1038/nbt.2280

[29] Antipov, D., Korobeynikov, A., McLean, J. S., & Pevzner, P. A. (2016). hybridSPAdes: analgorithm for hybrid assembly of short and long reads. Bioinformatics, 32(7), 1009–1015. http://doi.org/10.1093/bioinformatics/btv688

[30] Bashir, A., Klammer, A., Robins, W. P., Chin, C.-S., Webster, D., Paxinos, E., … Schadt, E. E. (2012). A Hybrid Approach for the Automated Finishing of Bacterial Genomes. Nature Biotechnology, 30(7), 701–707. http://doi.org/10.1038/nbt.2288

[31] http://i.cs.hku.hk/~alse/hkubrg/projects/idba_hybrid/index.html

[32] Antipov, D., Korobeynikov, A., McLean, J. S., & Pevzner, P. A. (2016). hybridSPAdes: analgorithm for hybrid assembly of short and long reads. Bioinformatics, 32(7), 1009–1015. http://doi.org/10.1093/bioinformatics/btv688

[33] Gurevich, A., Saveliev, V., Vyahhi, N., & Tesler, G. (2013). QUAST: quality assessment tool for genome assemblies. Bioinformatics, 29(8), 1072–1075. http://doi.org/10.1093/bioinformatics/btt086

[34] Utturkar, S. M., Klingeman, D. M., Land, M. L., Schadt, C. W., Doktycz, M. J., Pelletier, D. A., & Brown, S. D. (2014). Evaluation and validation of de novoand hybrid assembly techniques to derive high-quality genome sequences. Bioinformatics, 30(19), 2709–2716. http://doi.org/10.1093/bioinformatics/btu391

[35] Camacho, C., Coulouris, G., Avagyan, V., Ma, N., Papadopoulos, J., Bealer, K., & Madden, T. L. (2009). BLAST+: architecture and applications. *BMC Bioinformatics*, *10*, 421. http://doi.org/10.1186/1471-2105-10-421

[36] Katoh, K., Misawa, K., Kuma, K., & Miyata, T. (2002). MAFFT: a novel method for rapid multiple sequence alignment based on fast Fourier transform. Nucleic Acids Research, 30(14), 3059–3066.

[37] http://cube.univie.ac.at/gepard

[38] Sommer, D. D., Delcher, A. L., Salzberg, S. L., & Pop, M. (2007). Minimus: a fast, lightweight genome assembler. BMC Bioinformatics, 8, 64.

[39] http://mummer.sourceforge.net/MUMmer2.pdf

[40] http://last.cbrc.jp/

[41] Nadalin, F., Vezzi, F., & Policriti, A. (2012). GapFiller: a de novo assembly approach to fill the gap within paired reads. BMC Bioinformatics, 13(Suppl 14), S8. http://doi.org/10.1186/1471-2105-13-S14-S8

[42] http://soap.genomics.org.cn/soapdenovo.html

[43] Boetzer, M., Henkel, C. V., Jansen, H. J., Butler, D., & Pirovano, W. (2011). Scaffolding pre-assembled contigs using SSPACE. Bioinformatics. 27(4), 578–579 http://doi.org/10.1093/bioinformatics/btq683

[44] Thorvaldsdóttir, H., Robinson, J. T., & Mesirov, J. P. (2013). Integrative Genomics Viewer (IGV): high-performance genomics data visualization and exploration. Briefings in Bioinformatics, 14(2), 178–192. http://doi.org/10.1093/bib/bbs017

[45] Weirather, J. L., de Cesare, M., Wang, Y., Piazza, P., Sebastiano, V., Wang, X.-J., … Au, K. F. (2017). Comprehensive comparison of Pacific Biosciences and Oxford Nanopore Technologies and their applications to transcriptome analysis. *F1000Research*, *6*, 100. http://doi.org/10.12688/f1000research.10571.2

[46] Laehnemann, D., Borkhardt, A., & McHardy, A. C. (2016). Denoising DNA deep sequencing data—high-throughput sequencing errors and their correction. Briefings in Bioinformatics, 17(1), 154–179. http://doi.org/10.1093/bib/bbv029

[47] Bruce J. Walker, Thomas Abeel, Terrance Shea, Margaret Priest, Amr Abouelliel, Sharadha Sakthikumar, Christina A. Cuomo, Qiandong Zeng, Jennifer Wortman, Sarah K. Young, Ashlee M. Earl (2014) Pilon: An Integrated Tool for Comprehensive Microbial Variant Detection and Genome Assembly Improvement. PLoS ONE 9(11): e112963. doi:10.1371/journal.pone.0112963

[48] https://samtools.github.io/hts-specs/SAMv1.pdf

[49] https://github.com/douglasgscofield/PacBio-utilities

[50] Olson, N. D., Lund, S. P., Colman, R. E., Foster, J. T., Sahl, J. W., Schupp, J. M., … Zook, J. M. (2015). Best practices for evaluating single nucleotide variant calling methods for microbial genomics. *Frontiers in Genetics*, *6*, 235. http://doi.org/10.3389/fgene.2015.00235

[51] https://github.com/ekg/freebayes

[52] Langmead, B., & Salzberg, S. L. (2012). Fast gapped-read alignment with Bowtie 2. Nature Methods, 9(4), 357–359. http://doi.org/10.1038/nmeth.1923

[53] Chaisson, M. J., & Tesler, G. (2012). Mapping single molecule sequencing reads using basic local alignment with successive refinement (BLASR): application and theory. BMC Bioinformatics, 13, 238.

[54] http://bedtools.readthedocs.io/en/latest/content/tools/genomecov.html

[55] http://www.gnuplot.info/

[56] Liu, S., Huang, S., Rao, J., Ye, W., The Genome Denmark Consortium, Krogh, A., & Wang, J. (2015). Discovery, genotyping and characterization of structural variation and novel sequence at single nucleotide resolution from de novo genome assemblies on a population scale. GigaScience, 4, 64. http://doi.org/10.1186/s13742-015-0103-4

[57] Nattestad, M., & Schatz, M. C. (2016). Assemblytics: A web analytics tool for the detection of variants from an assembly. Bioinformatics, 32(19), 3021-3023. doi:10.1093/bioinformatics/btw369

[58] https://genome.ucsc.edu/FAQ/FAQformat.html#format1

[59] Peacock, C. S., Seeger, K., Harris, D., Murphy, L., Ruiz, J. C., Quail, M. A., … Berriman, M. (2007). Comparative genomic analysis of three *Leishmania* species that cause diverse human disease. *Nature Genetics*, *39*(7), 839–847. http://doi.org/10.1038/ng2053

[60] Steinbiss, S., Silva-Franco, F., Brunk, B., Foth, B., Hertz-Fowler, C., Berriman, M., & Otto, T. D. (2016). *Companion*: a web server for annotation and analysis of parasite genomes. *Nucleic Acids Research*, *44*(Web Server issue), W29–W34. http://doi.org/10.1093/nar/gkw292

[61] Otto, T. D., Dillon, G. P., Degrave, W. S., & Berriman, M. (2011). RATT: Rapid Annotation Transfer Tool. *Nucleic Acids Research*, *39*(9), e57. http://doi.org/10.1093/nar/gkq1268

[62] Korf, I. (2004). Gene finding in novel genomes. *BMC Bioinformatics*, *5*, 59. http://doi.org/10.1186/1471-2105-5-59

[63] Stanke, M., Schöffmann, O., Morgenstern, B., & Waack, S. (2006). Gene prediction in eukaryotes with a generalized hidden Markov model that uses hints from external sources. *BMC Bioinformatics*, *7*, 62. http://doi.org/10.1186/1471-2105-7-62

[64] Laslett, D., & Canback, B. (2004). ARAGORN, a program to detect tRNA genes and tmRNA genes in nucleotide sequences. Nucleic Acids Research, 32(1), 11–16. http://doi.org/10.1093/nar/gkh152

[65] Nawrocki, E. P., & Eddy, S. R. (2013). Infernal 1.1: 100-fold faster RNA homology searches. *Bioinformatics*, *29*(22), 2933–2935. http://doi.org/10.1093/bioinformatics/btt509

[66] Li, L. (2003). OrthoMCL: Identification of Ortholog Groups for Eukaryotic Genomes. *Genome Research, 13*(9), 2178-2189. doi:10.1101/gr.1224503

[67] Finn, R. D., Bateman, A., Clements, J., Coggill, P., Eberhardt, R. Y., Eddy, S. R., … Punta, M. (2014). Pfam: the protein families database. *Nucleic Acids Research*, *42*(Database issue), D222–D230. http://doi.org/10.1093/nar/gkt1223

[68] http://gmod.org/wiki/GFF3

[69] https://www.ebi.ac.uk/ena

[70] https://filezilla-project.org/

[71] http://eprints.ucm.es/13062/1/TFM-IA-DanielaDiasXavier.pdf

[72] https://stepik.org/lesson/24328/step/3

[73] Burge, S., Parkinson, G. N., Hazel, P., Todd, A. K., & Neidle, S. (2006). Quadruplex DNA: sequence, topology and structure. Nucleic Acids Research, 34(19), 5402–5415. http://doi.org/10.1093/nar/gkl655

[74] Zhou, Y., Bizzaro, J. W., & Marx, K. A. (2004). Homopolymer tract length dependent enrichments in functional regions of 27 eukaryotes and their novel dependence on the organism DNA (G+C) % composition. BMC Genomics, 5, 95. http://doi.org/10.1186/1471-2164-5-95

[75]https://translate.google.es/translate?hl=es&sl=en&u=https://en.wikipedia.org/wiki/K-mer&prev=search

[76] https://es.wikipedia.org/wiki/MD5

[77] Gu, W., Zhang, F., & Lupski, J. R. (2008). Mechanisms for human genomic rearrangements. PathoGenetics, 1, 4. http://doi.org/10.1186/1755-8417-1-4

8. Anexos

8.1 Anexo I

Tabla de cromosomas extendidos mediante diferentes métodos.

Cromosoma	Método
01	Extremo inicial. Contig Illumina. BLAST
03	Extremo final. Contig Illumina. Minimus2
04	Extremo inicial. Contig Illumina. BLAST
05	Extremo inicial. Contig Illumina. BLAST
06	Extremo inicial. Contig Illumina. MAFFT
07	Extremo inicial. Contig Illumina. BLAST
08	Extremo inicial. Contig Illumina. BLAST
09	Extremo inicial. Contig Illumina. LAST
10	Extremo inicial y final. Minimus2. GapFiller.
11	Extremo inicial. Contig Illumina. BLAST
12	Extremo final. Contig Illumina. MAFFT
13	Extremo final. Contig Illumina. BLAST
14	Extremo inicial. Contig Illumina. BLAST
15	Extremo inicial. Varios contig Illumina. LAST
16	Extremo inicial. Contig Illumina. BLAST. GapCloser.
18	Extremo inicial. Contig Illumina. BLAST. GapFiller.
21	Extremo inicial. Contig Illumina. BLAST
22	Extremo inicial. Contig Illumina. LAST
23	Extremo final. Contig Illumina. BLAST
24	Extremo inicial. Contig Illumina. MAFTT
26	Extremo inicial. Contig Illumina. BLAST
27	Extremo inicial. Contig Illumina. MAFTT
28	Extremo inicial. Contig Illumina. BLAST
29	Extremo inicial. Contig Illumina. SSPACE-standard
30	Extremo inicial. Contig Illumina. BLAST.
32	Extremo inicial. Contig Illumina. BLAST. GapCloser.
34	Extremo inicial. Contig Illumina. SSPACE-standard y GapFiller

8.2 Anexo II.

Código Gnuplot.

```
# Multiplot setting chr LmjF fragmented
# ############################
set terminal png truecolor size 1633,1155 large
set output 'all_2raw.ps'
set multiplot layout 3, 2
# LmjF.08
set title "LmjF.08"
set style data lines
set style fill transparent solid 0.5 noborder
unset key
set arrow from 0,17 to 295671,17 heads back filled linetype 1 linecolor rgb
    "black" linewidth 2.500
set arrow from 285837,16 to 526451,16 heads back filled linetype 1 linecolor rgb
    "black" linewidth 2.500
set label 'c08_1' at 100000,18
set label 'c08_2' at 360000,17
set xrange [0:534283]
set yrange [0:22]
set xlabel "Position (kb)"
set ylabel "Depth (log2)"
set xtics border out offset 0,0.5 0,100000,10000000
plot 'LmjFNew_08_reads_illumina_smooth_log2W200.dat' using 2 w lines  lc rgb
    "blue" , 'LmjFNew_08_reads_pacbio_smooth_log2W200.dat' using 2 w lines lw 2
    lc rgb "red"
# LmjF.19
unset arrow
unset label
set arrow from 0,17 to 434787,17 heads back filled linetype 1 linecolor rgb
    "black" linewidth 2.500
set arrow from 434761,16 to 728842,16 heads back filled linetype 1 linecolor rgb
    "black" linewidth 2.500
set label 'c19_1' at 100000,18
set label 'c19_2' at 515000,17
set title " LmjF.19"
set style data lines
set style fill transparent solid 0.5 noborder
unset key
set xrange [0:728967]
set yrange [0:22]
set xlabel "Position (kb)"
set ylabel "Depth (Log2)"
set xtics border out offset 0,0.5 0,100000,10000000
```

```
plot 'LmjFNew_19_reads_illumina_smooth_log2W200.dat' using 2 w lines  lc rgb
    "blue" , 'LmjFNew_19_reads_pacbio_smooth_log2W200.dat' using 2 w lines lw 2
    lc rgb "red"

# LmjF.22

unset arrow

unset label

set arrow from 0,17 to 265042,17 heads back filled linetype 1 linecolor rgb
    "black" linewidth 2.500

set arrow from 255126,16 to 690509,16 heads back filled linetype 1 linecolor rgb
    "black" linewidth 2.500

set label 'c22_1' at 100000,18

set label 'c22_2' at 515000,17

set title " LmjF.22"

set style data lines

set style fill transparent solid 0.5 noborder

unset key

set xrange [0:699913]

set yrange [0:22]

set xlabel "Position (kb)"

set ylabel "Depth (log2)"

set xtics border out offset 0,0.5 0,100000,10000000

plot 'LmjFNew_22_reads_illumina_smooth_log2W200.dat' using 2 w lines  lc rgb
    "blue" , 'LmjFNew_22_reads_pacbio_smooth_log2W200.dat' using 2 w lines lw 2
    lc rgb "red"

# LmjF.27

unset arrow

unset label

set arrow from 0,16 to 1007060,16 heads back filled linetype 1 linecolor rgb
    "black" linewidth 2.500

set arrow from 1006411,16 to 1095824,16 heads back filled linetype 1 linecolor
    rgb "black" linewidth 2.500

set label 'c27_1' at 400000,17

set label 'c27_2' at 1050000,17

set title "LmjF.27"

set style data lines

set style fill transparent solid 0.5 noborder

unset key

set xrange [0:1095858]

set yrange [0:22]

set xlabel "Position (kb)"

set ylabel "Depth (log2)"

set xtics border out offset 0,0.5 0,100000,10000000

plot 'LmjFNew_27_reads_illumina_smooth_log2W200.dat' using 2 w lines  lc rgb
    "blue" , 'LmjFNew_27_reads_pacbio_smooth_log2W200.dat' using 2 w lines lw 2
    lc rgb "red"

# LmjF.35

unset arrow
```

```
unset label
set arrow from 0,16 to 161993,16 heads back filled linetype 1 linecolor rgb
    "black" linewidth 2.500
set arrow from 161853,16 to 2032955,16 heads back filled linetype 1 linecolor
    rgb "black" linewidth 2.500
set label 'c35_1' at 700000,17
set label 'c35_2' at 50000,17
set title " LmjF.35 "
set style data lines
set style fill transparent solid 0.5 noborder
unset key
set xrange [0:2033137]
set yrange [0:22]
set xlabel "Position (kb)"
set ylabel "Depth (log2)"
set xtics border out offset 0,0.5 0,100000,10000000
plot 'LmjFNew_35_reads_illumina_smooth_log2W200.dat' using 2 w lines  lc rgb
    "blue" , 'LmjFNew_35_reads_pacbio_smooth_log2W200.dat' using 2 w lines lw 2
    lc rgb "red"


# Multiplot setting chr LmjF fragmented
# ############################
set terminal png truecolor size 1633,1155 large
set output 'all_raw_reference.ps'
set multiplot layout 3, 2
# LmjF.08
set title "LmjF.08"
set style data lines
set style fill transparent solid 0.5 noborder
unset key
set xrange [0:574960]
set yrange [0:22]
set xlabel "Position (kb)"
set ylabel "Depth (log2)"
set xtics border out offset 0,0.5 0,100000,10000000
plot 'LmjF_08_reads_illumina_smooth_log2W200.dat' using 2 w lines  lc rgb "blue"
    , 'LmjF_08_reads_pacbio_smooth_log2W200.dat' using 2 w lines lw 2 lc rgb
    "red"
# LmjF.19
unset arrow
unset label
set title " LmjF.19"
set style data lines
set style fill transparent solid 0.5 noborder
unset key
```

```
set xrange [0:706945]

set yrange [0:22]

set xlabel "Position (kb)"

set ylabel "Depth (Log2)"

set xtics border out offset 0,0.5 0,100000,10000000

plot 'LmjF_19_reads_illumina_smooth_log2W200.dat' using 2 w lines  lc rgb "blue"
    , 'LmjF_19_reads_pacbio_smooth_log2W200.dat' using 2 w lines lw 2 lc rgb
    "red"

# LmjF.22

unset arrow

unset label

set title " LmjF.22"

set style data lines

set style fill transparent solid 0.5 noborder

unset key

set xrange [0:716602]

set yrange [0:22]

set xlabel "Position (kb)"

set ylabel "Depth (log2)"

set xtics border out offset 0,0.5 0,100000,10000000

plot 'LmjF_22_reads_illumina_smooth_log2W200.dat' using 2 w lines  lc rgb "blue"
    , 'LmjF_22_reads_pacbio_smooth_log2W200.dat' using 2 w lines lw 2 lc rgb
    "red"

# LmjF.27

unset arrow

unset label

set title "LmjF.27"

set style data lines

set style fill transparent solid 0.5 noborder

unset key

set xrange [0:1134137]

set yrange [0:22]

set xlabel "Position (kb)"

set ylabel "Depth (log2)"

set xtics border out offset 0,0.5 0,100000,10000000

plot 'LmjF_27_reads_illumina_smooth_log2W200.dat' using 2 w lines  lc rgb "blue"
    , 'LmjF_27_reads_pacbio_smooth_log2W200.dat' using 2 w lines lw 2 lc rgb
    "red"

# LmjF.35

unset arrow

unset label

set title " LmjF.35 "

set style data lines

set style fill transparent solid 0.5 noborder

unset key
```

```
set xrange [0:2090474]
set yrange [0:22]
set xlabel "Position (kb)"
set ylabel "Depth (log2)"
set xtics border out offset 0,0.5 0,100000,10000000
plot 'LmjF_35_reads_illumina_smooth_log2W200.dat' using 2 w lines  lc rgb "blue"
    , 'LmjF_35_reads_pacbio_smooth_log2W200.dat' using 2 w lines lw 2 lc rgb
    "red"
```

8.3 Anexo III
Script en Python para detectar longitudes

```
#!/usr/bin/env python
# encoding:UTF-8

# Imports
from __future__ import division
from collections import defaultdict
import sys
import os

syntax = '''-----------------------------------------------------------------------
    ----------

    Descripcion: El script recibe una tabla. Hace calculos y genera una nueva
    tabla con nuevo end y longitud de un archivo BED.
    Elimina aquellas variantes que cuya longitud sea menor a 100
    python script.py bed-file output_file
-------------------------------------------------------------------------------
    '''
if len(sys.argv) != 3:
        print syntax
        sys.exit()

#--Parameters
bed_1 = sys.argv[1]
output_file = sys.argv[2]


inhandle1 = open(bed_1 , 'r')
outhandle = open(output_file , 'w')

# Main program
lines1 = inhandle1.readlines()
length_old = 0
```

```
sumaTot = 0
for n in range(len(lines1)):
    l1 = lines1[n]
    line1 = l1.rstrip("\n")
    chunks1 = line1.split("\t")
    inicio = int(chunks1[1])
    final = int(chunks1[2])
    genes = chunks1[3]
    length = ((final - inicio))
    if length >= 100:
      data = line1 +  "\t" + str(length) + "\n"
      outhandle.write(data)
```

8.4 Anexo IV
Script en Python para generar un archivo gff con información de otros programas.

```
#!/usr/bin/env python
# Imports
from __future__ import division
from collections import defaultdict
import sys, os
import math

syntax = '''-------------------------------------------------------------------------
    ----------


        Descripcion: Script que a partir de un archivo de dos columnas (id +
    orthoMCL id) mete el
        gen ortologo en el gff y lo exporta a un nuevo gff (LEISHMANIA MAJOR).
    Companion.
-------------------------------------------------------------------------------
    '''


list_orthologos = sys.argv[1]
gff3 = sys.argv[2]
output_file = sys.argv[3]


outhandle = open(output_file , 'w')
inhandle1 = open(list_orthologos , 'r')
inhandle2 = open( gff3 , 'r')


# lee el archivo de la lista de genes con sus ortologos.
dict_orthologos = {}
lines1 = inhandle1.readlines()
```

```
for line1 in lines1:
        chunk = line1.split("\t")
        id_infantumNew = chunk[0]
        id_infantumNew = id_infantumNew[:-2]
        id_infantumRef = chunk[2]
        id_infantumRef = id_infantumRef.split("|")
        id_infantumRef = id_infantumRef[1]
        dict_orthologos[id_infantumNew] = id_infantumRef

inhandle1.close()

# Lee el documento gff3.
lines2 = inhandle2.readlines()
for line2 in lines2:
  line2 = line2.rstrip("\r\n")
  if "#" in line2:
    outhandle.write(line2 + "\n")
  else:
      chunk2 = line2.split("\t")
      if chunk2[2] == "gene" or chunk2[2] == "pseudogene" or chunk2[2]
    =="polypeptide" or chunk2[2] == "mRNA":
        id_gff = chunk2[8]
        id_gff = id_gff.split(";")
        id_gff = id_gff[0].split("=")
        id_gff = id_gff[1].split(":")
        id_gff = id_gff[0]
        id_gff = id_gff.split(".")
        id_gff = id_gff[0]
        try:
           outhandle.write(line2 + ";" + "ORTHOMLC:"+ dict_orthologos[id_gff]+
    "\n")
        except KeyError:
           outhandle.write(line2 + ";" + "ORTHOMLC_blast not found"+ "\n")
      else:
        outhandle.write(line2 + "\n")
```

8.5 Tabla suplementaria S1

Cromosoma	inicio	fin	Gen Afectado
LmjF.02	281849	282113	LmjF.02.SLRNA.0270
LmjF.02	282113	282559	LmjF.02.SLRNA.0280
LmjF.02	282559	283001	LmjF.02.SLRNA.0290
LmjF.02	283001	283450	LmjF.02.SLRNA.0300
LmjF.02	283450	283896	LmjF.02.SLRNA.0310
LmjF.02	283896	284342	LmjF.02.SLRNA.0320
LmjF.02	284342	284797	LmjF.02.SLRNA.0330
LmjF.02	284797	285242	LmjF.02.SLRNA.0340
LmjF.02	285242	285694	LmjF.02.SLRNA.0350
LmjF.02	285694	286149	LmjF.02.SLRNA.0360
LmjF.02	286149	286604	LmjF.02.SLRNA.0370
LmjF.02	286604	287059	LmjF.02.SLRNA.0380
LmjF.02	287059	287517	LmjF.02.SLRNA.0390
LmjF.02	287517	287967	LmjF.02.SLRNA.0400
LmjF.02	287967	288416	LmjF.02.SLRNA.0410
LmjF.02	288416	288873	LmjF.02.SLRNA.0420
LmjF.02	288874	289323	LmjF.02.SLRNA.0430
LmjF.02	289323	289772	LmjF.02.SLRNA.0440
LmjF.02	289772	290227	LmjF.02.SLRNA.0450
LmjF.02	290227	290681	LmjF.02.SLRNA.0460
LmjF.02	290681	291129	LmjF.02.SLRNA.0470
LmjF.02	291129	291583	LmjF.02.SLRNA.0480
LmjF.02	291583	292032	LmjF.02.SLRNA.0490
LmjF.02	292032	292478	LmjF.02.SLRNA.0500
LmjF.02	292478	292936	LmjF.02.SLRNA.0510
LmjF.02	292936	293378	LmjF.02.SLRNA.0520

Cromosoma	inicio	fin	Gen Afectado
LmjF.02	293378	293833	LmjF.02.SLRNA.0530
LmjF.02	293833	294288	LmjF.02.SLRNA.0540
LmjF.02	294288	294737	LmjF.02.SLRNA.0550
LmjF.02	294737	294921	LmjF.02.SLRNA.0560
LmjF.05	113967	114083	LmjF.05.0380
LmjF.05	449127	449202	LmjF.05.snoRNA0075
LmjF.05	449230	449299	LmjF.05.snoRNA0090
LmjF.05	449645	449719	LmjF.05.snoRNA0085
LmjF.05	449811	449907	LmjF.05.snoRNA0080
LmjF.05	450012	450297	LmjF.05.snoRNA0072
LmjF.05	450321	450390	LmjF.05.snoRNA0068
LmjF.05	450419	450512	LmjF.05.snoRNA0064
LmjF.05	450594	450669	LmjF.05.snoRNA0076
LmjF.05	450697	450766	LmjF.05.snoRNA0091
LmjF.05	451112	451186	LmjF.05.snoRNA0086
LmjF.05	451278	451374	LmjF.05.snoRNA0081
LmjF.05	451479	451764	LmjF.05.snoRNA0073
LmjF.05	451788	451857	LmjF.05.snoRNA0069
LmjF.05	451886	451979	LmjF.05.snoRNA0065
LmjF.05	452061	452136	LmjF.05.snoRNA0077
LmjF.05	452164	452233	LmjF.05.snoRNA0092
LmjF.08	321866	322481	LmjF.08.0730
LmjF.08	380915	381497	LmjF.08.0840
LmjF.08	467200	468247	LmjF.08.1040
LmjF.09	73871	74031	LmjF.09.0162
LmjF.09	75131	75506	LmjF.09.0164
LmjF.09	76688	77093	LmjF.09.0166
LmjF.10	216395	218204	LmjF.10.0465
LmjF.11	18530	18807	LmjF.11.0070
LmjF.13	109971	111327	LmjF.13.0360

Cromosoma	inicio	fin	Gen Afectado
LmjF.14	484299	484930	LmjF.14.1120
LmjF.16	613912	614856	LmjF.16.1460
LmjF.16	706763	707202	LmjF.16.1660
LmjF.16	707309	707534	LmjF.16.1660
LmjF.16	708159	708790	LmjF.16.1660
LmjF.16	703631	706700	LmjF.16.1660
LmjF.18	539047	539755	LmjF.18.1275
LmjF.18	541789	542034	LmjF.18.1280
LmjF.19	368488	368866	LmjF.19.0870
LmjF.19	369556	369922	LmjF.19.0880
LmjF.19	371990	372368	LmjF.19.0900
LmjF.19	373053	373419	LmjF.19.0910
LmjF.22	587395	587468	LmjF.22.snoRNA0036
LmjF.22	587517	587616	LmjF.22.snoRNA0030
LmjF.22	587707	587740	LmjF.22.snoRNA0040
LmjF.25	387712	388722	LmjF.25.1000
LmjF.25	391289	392585	LmjF.25.1010
LmjF.25	393783	393993	LmjF.25.1015
LmjF.25	395599	396895	LmjF.25.1020
LmjF.25	399439	401716	LmjF.25.1030
LmjF.25	403993	406195	LmjF.25.1040
LmjF.25	407233	407682	LmjF.25.1050
LmjF.25	618265	618356	LmjF.25.snoRNA0140
LmjF.25	618441	618555	LmjF.25.snoRNA0126
LmjF.25	618616	618694	LmjF.25.snoRNA0168
LmjF.25	618763	618846	LmjF.25.snoRNA0152
LmjF.25	619292	619370	LmjF.25.snoRNA0169
LmjF.25	619439	619522	LmjF.25.snoRNA0153
LmjF.25	619616	619706	LmjF.25.snoRNA0017
LmjF.25	619791	619905	LmjF.25.snoRNA0128

Cromosoma	inicio	fin	Gen Afectado
LmjF.25	619966	620032	LmjF.25.snoRNA0019
LmjF.25	620143	620226	LmjF.25.snoRNA0154
LmjF.25	620320	620410	LmjF.25.snoRNA0021
LmjF.25	620496	620609	LmjF.25.snoRNA0138
LmjF.25	620670	620748	LmjF.25.snoRNA0170
LmjF.25	620817	620900	LmjF.25.snoRNA0155
LmjF.25	620995	621086	LmjF.25.snoRNA0142
LmjF.25	621171	621285	LmjF.25.snoRNA0129
LmjF.25	621346	621424	LmjF.25.snoRNA0171
LmjF.25	621493	621576	LmjF.25.snoRNA0156
LmjF.25	621670	621760	LmjF.25.snoRNA0029
LmjF.25	621845	621959	LmjF.25.snoRNA0130
LmjF.25	622020	622098	LmjF.25.snoRNA0172
LmjF.25	622167	622250	LmjF.25.snoRNA0157
LmjF.25	622345	622436	LmjF.25.snoRNA0143
LmjF.25	622522	622632	LmjF.25.snoRNA0034
LmjF.25	622696	622774	LmjF.25.snoRNA0173
LmjF.25	622843	622926	LmjF.25.snoRNA0158
LmjF.25	623020	623110	LmjF.25.snoRNA0037
LmjF.25	623195	623309	LmjF.25.snoRNA0131
LmjF.26	880869	881157	LmjF.26.2170
LmjF.27	57137	57380	LmjF.27.0240
LmjF.27	152703	154132	LmjF.27.0490
LmjF.27	376538	376607	LmjF.27.snoRNA0162
LmjF.27	376676	376767	LmjF.27.snoRNA0159
LmjF.27	376878	376934	LmjF.27.snoRNA0179
LmjF.27	376975	377065	LmjF.27.snoRNA0128
LmjF.27	377138	377207	LmjF.27.snoRNA0163
LmjF.27	377276	377367	LmjF.27.snoRNA0152
LmjF.27	377447	377533	LmjF.27.snoRNA0174

Cromosoma	inicio	fin	Gen Afectado
LmjF.27	377574	377664	LmjF.27.snoRNA0129
LmjF.27	377709	377778	LmjF.27.snoRNA0146
LmjF.27	377899	377965	LmjF.27.snoRNA0141
LmjF.27	378048	378138	LmjF.27.snoRNA0130
LmjF.27	378211	378280	LmjF.27.snoRNA0164
LmjF.27	378349	378440	LmjF.27.snoRNA0153
LmjF.27	378521	378607	LmjF.27.snoRNA0180
LmjF.27	378648	378738	LmjF.27.snoRNA0131
LmjF.27	378811	378880	LmjF.27.snoRNA0165
LmjF.27	378949	379040	LmjF.27.snoRNA0154
LmjF.27	379121	379152	LmjF.27.snoRNA0175
LmjF.27	380795	380881	LmjF.27.snoRNA0182
LmjF.27	380922	381012	LmjF.27.snoRNA0135
LmjF.27	381085	381154	LmjF.27.snoRNA0169
LmjF.27	381223	381314	LmjF.27.snoRNA0156
LmjF.27	1005252	1007456	LmjF.27.rRNA.02
LmjF.27	1007599	1007882	LmjF.27.rRNA.08
LmjF.27	1008549	1010329	LmjF.27.rRNA.14
LmjF.27	1010417	1010630	LmjF.27.rRNA.20
LmjF.27	1010954	1012479	LmjF.27.rRNA.26
LmjF.27	1012510	1012693	LmjF.27.rRNA.32
LmjF.27	1013237	1013310	LmjF.27.rRNA.38
LmjF.27	1013561	1013690	LmjF.27.rRNA.45
LmjF.27	1014084	1014213	LmjF.27.rRNA.46
LmjF.28	1152039	1153136	LmjF.28.3030
LmjF.29	421184	421584	LmjF.29.1070
LmjF.29	422844	423569	LmjF.29.1080
LmjF.31	357383	359249	LmjF.31.0950
LmjF.33	142978	145084	LmjF.33.0333
LmjF.33	146610	148716	LmjF.33.0336

Cromosoma	inicio	fin	Gen Afectado
LmjF.33	343025	344357	LmjF.33.0794
LmjF.33	346608	347940	LmjF.33.0796
LmjF.33	350191	351523	LmjF.33.0798
LmjF.33	353777	355109	LmjF.33.0800
LmjF.33	357360	358692	LmjF.33.0802
LmjF.33	360947	362279	LmjF.33.0804
LmjF.33	389621	390953	LmjF.33.0819
LmjF.33	1495210	1495613	LmjF.33.3070
LmjF.33	1497009	1497157	LmjF.33.3070
LmjF.33	1506212	1506871	LmjF.33.3070
LmjF.33	1509315	1509626	LmjF.33.3070
LmjF.33	1511695	1512040	LmjF.33.3070
LmjF.34	315518	315814	LmjF.34.0690
LmjF.34	1148878	1149128	LmjF.34.2530
LmjF.34	1173172	1173522	LmjF.34.2560
LmjF.34	1176202	1176321	LmjF.34.2560
LmjF.34	1814239	1814902	LmjF.34.4340
LmjF.34	1815628	1815967	LmjF.34.4350
LmjF.34	1816551	1817241	LmjF.34.4360
LmjF.35	215627	244161	LmjF.35.0540
LmjF.35	256829	257020	LmjF.35.0550
LmjF.35	1675096	1675169	LmjF.35.snoRNA0186
LmjF.35	1675221	1675297	LmjF.35.snoRNA0167
LmjF.35	1676368	1676369	LmjF.35.snoRNA0176
LmjF.35	1676407	1676480	LmjF.35.snoRNA0187
LmjF.35	1676532	1676608	LmjF.35.snoRNA0168
LmjF.35	1680306	1680307	LmjF.35.snoRNA0179
LmjF.35	1680345	1680418	LmjF.35.snoRNA0190
LmjF.35	1680470	1680546	LmjF.35.snoRNA0171