Evaluación y comparación de métodos de ensamblaje y *binning* a partir de datos metagenómicos reales

Andrea Vergara Gómez

Trabajo Fin de Máster (TFM) - Máster Bioinformática y Bioestadística
Área 1.12: Genómica comparativa

Tutor IrsiCaixa: **Marc Noguera**; Tutora UOC: **Yolanda Guillén**
Profesor responsable de la asignatura: **Carles Ventura**

Fecha Entrega: 04/06/2019

FICHA DEL TRABAJO FINAL

Título del trabajo:	Evaluación y comparación de métodos de ensamblaje y *binning* a partir de datos metagenómicos reales
Nombre del autor:	Andrea Vergara Gómez
Nombre de la consultora:	Yolanda Guillén Montalbán
Nombre del PRA:	Javier Luis Cánovas Izquierdo
Fecha de entrega (mm/aaaa):	06/2019
Titulación:	Máster universitario en Bioinformática y bioestadística UOC-UB
Área del Trabajo Final:	Genómica comparativa
Idioma del trabajo:	Castellano
Palabras clave	Metagenómica, ensamblaje, *binning*

Resumen del Trabajo:

La secuenciación masiva ha permitido analizar el contenido genómico de todos los microorganismos de una muestra (metagenómica), sin necesidad de cultivarlos. El análisis de datos *shotgun* representa un gran reto. Agrupar las secuencias obtenidas en distintas especies metagenómicas basándose en referencias externas supone que muchas secuencias quedan sin asignar, por lo que parecen más adecuados los métodos independientes de referencia (*binning*). El objetivo de este trabajo fue comparar dos métodos de ensamblaje y dos métodos de *binning* analizando datos metagenómicos reales. Se realizó el ensamblaje *de-novo* de las secuencias depuradas con dos ensambladores: MEGAHIT y MetaSPAdes. La bondad de estos ensamblajes se analizó con QUAST. A partir de los *contigs*, se generó un catálogo de genes únicos y se realizó el *binning* con Canopy y MetaBAT2. La bondad de los *binning* se evaluó con CheckM. Se trabajó en un clúster de supercomputadores y, siempre que fue posible, los trabajos se ejecutaron en paralelo, para optimizar el tiempo de

análisis. En relación al ensamblaje, se obtuvieron mejores resultados utilizando MetaSPAdes que MEGAHIT. Respecto al *binning*, los resultados obtenidos indican que Canopy generó muchos más *bins* que MetaBAT2, pero al visualizar los *bins* obtenidos se comprobó que los resultados eran sub-óptimos para ambos. Trabajar en un clúster de PCs permite ahorrar tiempo de análisis y optimizar recursos. Teniendo en cuenta estos datos, son necesarios nuevos enfoques para conseguir mejores resultados: estrategia single-sample basada en *contigs*, usar contigs completos en lugar de genes y testear el resultado de co-ensamblaje múltiple para varias muestras.

Abstract:

Thanks to the Next Generation Sequencing it is possible to analyze the genes of all the microorganisms in a sample (metagenomics), without the need to cultivate them. The analysis of shotgun data represents a great challenge. Grouping sequences from different metagenomic species based on external references means that many sequences will remain unassigned, so it seems more appropriate to use the reference independent methods (binning). The objective of this study was to compare two assemblers and two binning methods with real metagenomic data. The *de-novo* assembly of trimmed reads was performed with two assemblers: MEGAHIT and MetaSPAdes. The performance of these assemblies was analyzed with QUAST. A catalog of unique genes was generated from the contigs and binning with Canopy and MetaBAT2 was carried out. The performance of the binning was evaluated with CheckM. A cluster of supercomputers was used and, whenever possible, jobs were executed in parallel, in order to optimize time of analysis. Regarding the

assembly, better results were obtained using MetaSPAdes than MEGAHIT. Regarding the binning, Canopy generated many more bins than MetaBAT2, but the visualization of the bins showed that the results were suboptimal for both. Working in a cluster of PCs allows you to save analysis time and optimize resources. According to these data, new approaches are necessary to achieve better results: the single-sample strategy based on contigs, using complete contigs instead of genes and testing the result of multiple co-assembly for several samples.

Índice

Agradecimientos

En primer lugar, me gustaría dar las gracias a todo el grupo de Genómica bacteriana de Irsicaixa por acogerme siempre tan bien. Son personas increíbles tanto a nivel profesional como personal. En especial, quería dar las gracias a Marc por todo lo que me ha enseñado y por su paciencia. Siempre ha tenido tiempo para resolver mis dudas y guiarme en el camino.

Gracias también a Yolanda por sus sugerencias y buenas palabras. Ha sido muy agradable trabajar con ella tanto en las prácticas como en el TFM. Gracias también a mi jefe por su confianza y darme la oportunidad de cursar este máster y perseguir mis metas. Y gracias a ti Efrén, compañero de aventura, por estar ahí, pero sobre todo por aguantarme en los malos momentos, en los que no veía la luz al final del túnel.

Lista de figuras

Lista de tablas

1. Introducción

1.1 Contexto del Trabajo

La secuenciación de alto rendimiento o secuenciación masiva ha permitido analizar el contenido genético de todos los microorganismos de una muestra de un determinado nicho ecológico, sin necesidad de cultivar esos microorganismos. El estudio de todos los genomas presentes en una muestra es lo que se conoce como metagenómica y se obtiene mediante secuenciación masiva de tipo *shotgun*. A diferencia de la secuenciación de genes marcadores para caracterizar la composición taxonómica de una muestra o metataxonomía, la metagenómica nos permite detectar microorganismos pertenecientes a todos los dominios (1), así como determinar y cuantificar todo el contenido genético de una muestra. En algunas ocasiones, el término metagenómica se utiliza indistintamente para secuenciación *shotgun* y secuenciación de genes marcadores.

De forma general, un estudio de metagenómica basado en secuenciación *shotgun* se compone de los siguientes pasos: recogida de muestras y extracción de ácidos nucleicos, preparación de librerías, secuenciación *shotgun*, preprocesado de las secuencias, análisis bioinformático de las secuencias, y análisis estadístico e interpretación biológica de los datos (Figura 1) (2-5).

Figura 1. Esquema general de un experimento de secuenciación masiva *shotgun*. El análisis primario consiste en transformar las imágenes generadas por el secuenciaciador en archivos del tipo fastq. Dentro del análisis secundario, quedan englobados todos los análisis realizados a partir de esos archivos fastq. El trabajo a desarrollar se engloba dentro del último paso (análisis secundario). AANN: Ácidos nucleicos.

Centrándonos en el análisis de las secuencias o *reads*, el primer paso comprende el análisis de calidad, mediante el cual se eliminan secuencias de baja calidad y posibles contaminantes (adaptadores, virus PhiX utilizado como control de secuenciación, etc). Después del control de calidad, las secuencias se ensamblan en secuencias más largas llamadas *contigs*. A partir de estos *contigs*, se pueden llegar a reconstruir los genomas presentes en las muestras. La asignación taxonómica se puede realizar a partir de los *reads* o de los *contigs* y permite conocer la composición de esas muestras (2,4,5). En la Figura 2 se muestra un ejemplo general del *pipeline* de análisis, aunque existen múltiples esquema posibles.

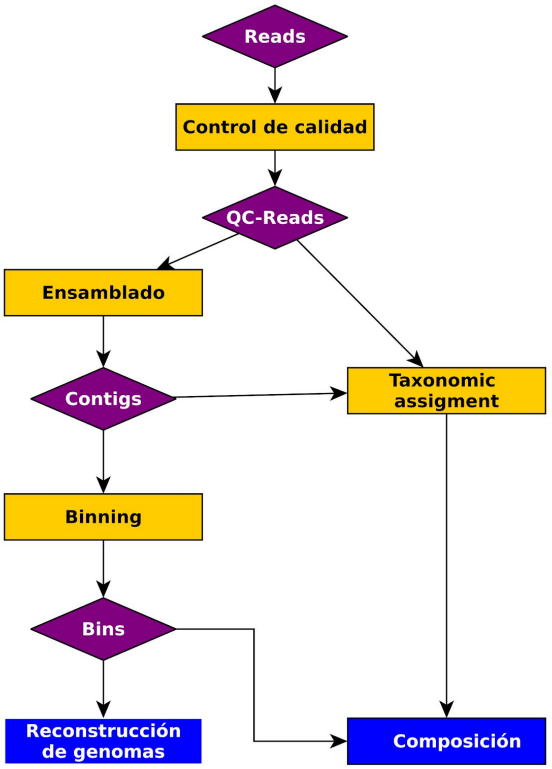

Figura 2. Esquema general del proceso de análisis de datos de secuenciación masiva *shotgun*. El trabajo a desarrollar se engloba dentro de los pasos de ensamblado y *binning*. QC-Reads: reads que han pasado el control de calidad.

Existen dos estrategias diferentes para llevar a cabo el proceso de cuantificación de especies microbianas (2,6). La primera de ellas es la dependiente de una referencia externa o clasificación taxonómica directa: los *reads* se clasifican taxonómicamente comparándolos con secuencias presentes en bases de datos de referencia (GenBank, EMBL o DDBJ, entre otras). La segunda es la no supervisada o *binning*, que requiere un paso previo de ensamblaje de los *reads* en secuencias más largas llamadas *contigs*. A partir de estos *contigs*, las secuencias se agrupan basándose en su composición y/o abundancia mediante diferentes algoritmos sin

necesidad de una referencia (7,8). Los métodos de *binning* suponen que la abundancia de genes de una misma especie covaría para un mismo taxón y/o que los contigs de un mismo *bin* tienen frecuencias de uso de k-meros (todas las posibles combinaciones de nucleótidos de longitud k que están contenidas en una secuencia) similares. De esta manera, los *contigs* se agrupan en *bins*. Los *reads* dentro de estos *bins* se pueden volver a ensamblar de nuevo, generando *contigs* de mejor calidad (menos y más largos) y cada grupo de secuencias se corresponde con una especie metagenómica diferente, que asumimos que son consistentes. Teóricamente, los genes de diferentes especies deberían ir a *bins* diferentes. Posteriormente, a estas especies metagenómicas se les puede asignar la taxonomía, anotar y cuantificar. Hasta este último paso, el proceso es totalmente independiente de cualquier referencia externa.

Existen los siguientes tipos de métodos de *binning*: basados en la composición de las secuencias, como MetaCluster (9) o MaxBin (10); basados en la co-abundancia de secuencias entre muestras, como Canopy (8) y GroopM (11); y mixtos, que tienen en cuenta tanto la composición como la abundancia, como CONCOCT (12) y MetaBAT (13).

1.2 Justificación del Trabajo

El análisis de datos metagenómicos representa un gran reto, entre otros motivos porque existen múltiples opciones de *pipeline* con aproximaciones variadas. Un paso fundamental por su complejidad y gran cantidad de opciones disponibles es el de agrupar las secuencias obtenidas en distintas especies metagenómicas. Los métodos dependientes de referencia presentan la gran desventaja de que las bases de datos son, por definición, incompletas, por lo que muchas secuencias quedan sin asignar o se asignan con baja resolución taxonómica. Por ello, parece más adecuado utilizar los métodos no supervisados o independientes de referencia. Sin embargo, dentro de este grupo, existen a su vez diferentes opciones tanto para el ensamblaje de los *reads* como para el *binning* de los *contigs*, fundamentales para un buen análisis de datos de metagenómica. Para el proceso de *binning,* se agrupan las secuencias en entes biológicos, sin basarse en una referencia externa. Esta agrupación se realiza de acuerdo a las características de las secuencias del propio experimento como las abundancias relativas y frecuencias de k-meros. Tampoco en este tipo de proceso hay un consenso sobre cuál utilizar.

Otro aspecto importante a considerar es que estamos trabajando con cantidades enormes de datos (cientos de gigabytes), por lo que procesar estos análisis no resulta una tarea sencilla. Resulta imprescindible para este tipo de análisis recurrir a sistemas de computación de alto rendimiento (Figura 3). Por otro lado, y no menos importante es el manejo y almacenamiento de los datos. Es fundamental contar con un sistema de computación de alto rendimiento adecuado donde almacenar y procesar tanto los datos crudos como todos los resultados de los

análisis realizados y además contar con un *backup* que nos permita disponer de copias de seguridad.

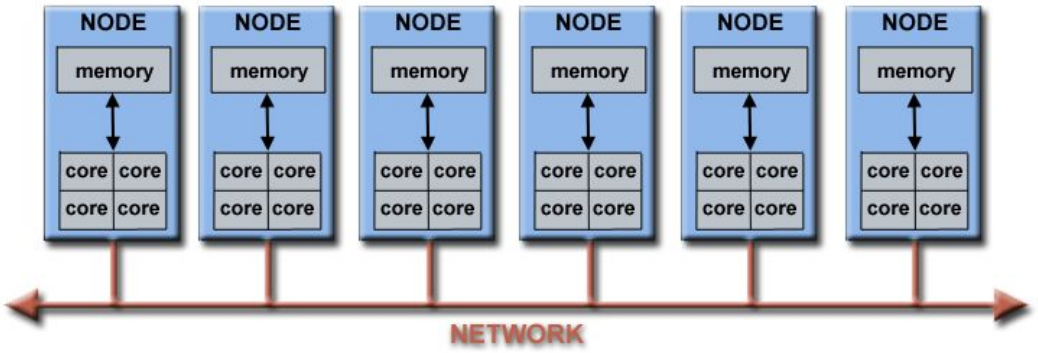

Figura 3. Clúster de PCs. Un ejemplo de sistema de computación de alto rendimiento es un clúster de PCs, un conjunto de ordenadores que está diseñado para dar altas prestaciones en cuanto a capacidad de cálculo. Varios PCs (nodos) pueden conectarse en red para conseguir sistemas de PCs en paralelo mayores.

Fuente: https://computing.llnl.gov/tutorials/parallel_comp/#Whatis

1.3 Motivación personal

Soy farmacéutica especialista en Microbiología Clínica, ámbito en el que la secuenciación masiva está ganando terreno desde hace unos años. Esto conlleva la generación de cantidades enormes de datos que hoy por hoy nos resulta difícil de manejar. Considero que la Bioinformática y la Bioestadística son fundamentales para mi futuro profesional, por lo que desde hace unos años decidí empezar a formarme en este campo, centrándome sobre todo en la secuenciación masiva.

Voy a realizar el TFM en el Instituto de Investigación del Sida IrsiCaixa, en concreto en el grupo de Genómica bacteriana, que tiene como investigador principal a Roger Paredes. Mi tutor es Marc Noguera, persona responsable del equipo y con gran experiencia en el campo de la Bioinformática y la Computación. Es el responsable del análisis de los datos, fundamentalmente análisis del microbioma. Mi tutora dentro de la UOC es Yolanda Guillén, especialista en genómica computacional, incluyendo el manejo de datos de secuenciación masiva, y que también formó parte del grupo de Genómica del IrsiCaixa como investigadora postdoc.

Como parte de este máster, realicé las prácticas en empresa con el mismo equipo, enfocando el trabajo en el análisis de la microbiota mediante amplificación del gen 16S rRNA. El siguiente escalón consiste en el análisis de datos *shotgun*. Este análisis representa un reto tanto a nivel computacional como de interpretación de los datos.

1.4 Objetivos del Trabajo

1.4.1 Objetivo general

Evaluar las opciones principales de *pipeline* de análisis de datos metagenómicos, comparando diferentes métodos de ensamblaje de secuencias y de *binning* de *contigs.*

1.4.2 Objetivos específicos

1. Establecer un *pipeline* de análisis de datos metagenómicos.
2. Aprender a trabajar en un clúster de ordenadores: conocer el funcionamiento y organización de un clúster de ordenadores y establecer las necesidades básicas a nivel computacional para realizar un análisis de metagenómica.
3. Comparar diferentes métodos de ensamblaje de secuencias.
4. Revisar las características de los métodos de *binning* y comparar el rendimiento de alguno de ellos.

1.5 Enfoque y método seguido

Para la realización del TFM se van a utilizar secuencias de un estudio ya publicado (14). Se trata de un estudio transversal que incluye pacientes infectados por el virus de la inmunodeficiencia humana tipo 1 (VIH-1) con distintas cargas virales y estado inmunológico, así como a pacientes no infectados por el VIH-1. Analizan el microbioma intestinal de todos estos pacientes y lo que observan es que el valor de CD4 más bajo alcanzado durante el seguimiento de los paciente con VIH-1, y no la presencia de infección por VIH-1, predice la disbiosis intestinal. Además, observan que los cambios en composición y funcionalidad del microbioma intestinal de estos pacientes VIH son muy parecidos a los encontrados en otras patologías que presentan también inflamación intestinal.

Para este estudio, el análisis de los datos se llevó a cabo siguiendo un *pipeline* que incluía la clasificación de los *reads* en función de una referencia externa. Lo que se pretende con este TFM analizar las mismas secuencias pero esta vez basándonos en métodos de agrupación de secuencias independientes de una referencia externa, para evaluar diferentes estrategias de análisis. El pipeline de análisis que se va a seguir se muestra en la Figura 4.

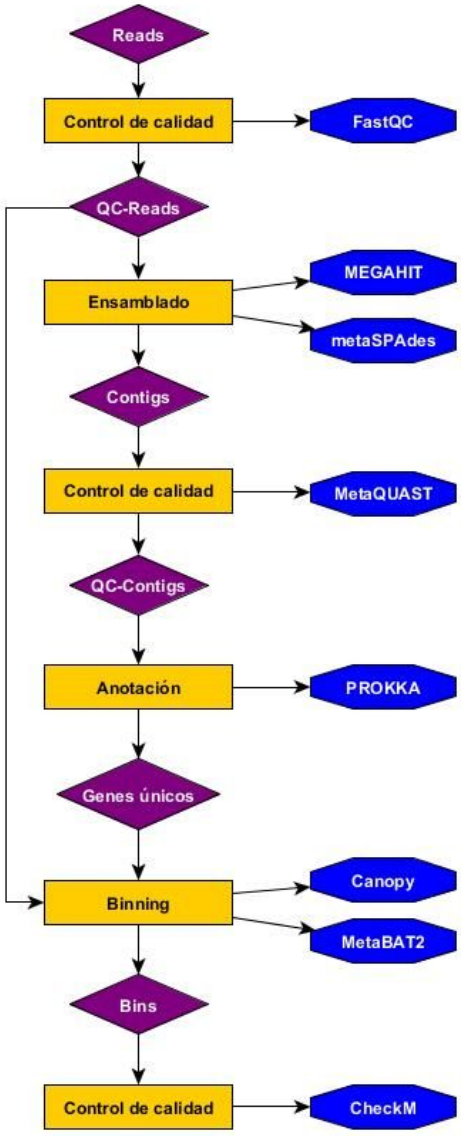

Figura 4. Esquema del proceso de análisis de datos que se va a seguir en el desarrollo del TFM. QC-Reads: reads que han pasado el control de calidad.

Lo que se pretende además es optimizar el análisis para conseguir obtener los resultados lo más rápido posible. Para ello, cada paso del *pipeline* se realizará primero con una única muestra en local, para luego pasar a lanzar el trabajo al clúster. Tras comprobar el *script* con con una única muestra, se trabajará con todo el

set de datos, primero usando un único core, **en serie**; y después **en paralelo**, usando varios cores de un único nodo y, finalmente, varios cores de diferentes nodos (Figura 5). En definitiva, lo que se consigue es resolver un problema en menos tiempo con múltiples recursos informáticos.

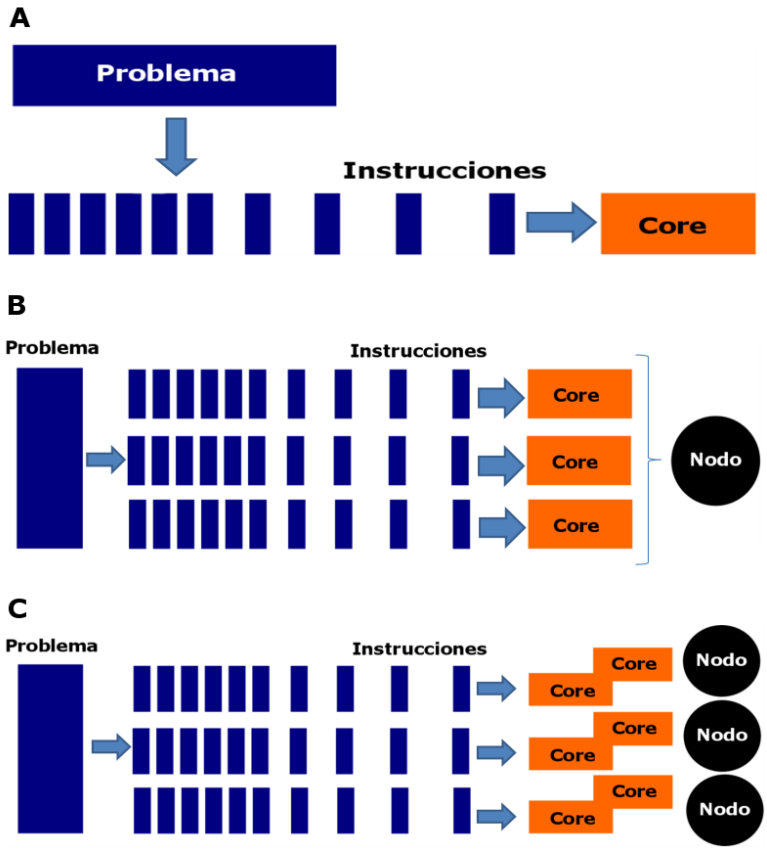

Figura 5. Diferencias entre procesamiento en serie (*serial computing*) y procesamiento en paralelo (*parallel computing*). En el procesamiento en serie (A), un problema se divide en una serie de instrucciones que se ejecutan secuencialmente una tras otra en un solo *core* (solo una instrucción puede ejecutarse cada vez). En el procesamiento en paralelo, un problema se divide en partes que pueden resolverse simultáneamente. Cada parte se desglosa en una serie de instrucciones que se ejecutan simultáneamente en diferentes cores de un mismo nodo (B) o de diferentes nodos (C).

1.6 Recursos necesarios

- **Software** para la realización de cada una de las distintas tareas. Todos los software que se van a utilizar en este TFM son de acceso libre, por lo que no será necesaria ninguna licencia.
- **Ordenador**. Para realizar los análisis es necesario tanto un ordenador personal como un ordenador en IrsiCaixa. Con estos ordenadores y mediante una VPN, se establecerá la conexión con el clúster.
- **Clúster**. IrsiCaixa proporciona acceso remoto un clúster mediante el cual podré realizar las diferentes tareas que requieren una alta capacidad computacional.
- **Datos metagenómicos**. Todos los análisis se realizarán con datos reales proporcionados por el centro que forman parte de un estudio ya publicado (14).

1.7 Planificación del Trabajo

1.7.1 Tareas

Objetivo 1. Establecer un pipeline de análisis de datos metagenómicos.

Tarea 1.1 Revisión bibliográfica.

Tarea 1.2 Establecer el *pipeline* de análisis para realizar el TFM.

Tarea 1.3 Crear un glosario con los términos propios del tema.

Objetivo 2. Aprender a trabajar en un clúster de ordenadores.

Tarea 2.1 Conocer el funcionamiento y organización del clúster de ordenadores empleado en IrsiCaixa.

Tarea 2.2 Establecer la conexión mediante VPN (red privada virtual) entre el clúster y el PC personal Windows, así como con el PC IrsiCaixa Linux.

Objetivo 3. Comparar diferentes métodos de ensamblaje de secuencias.

Tarea 3.1 Analizar la calidad de las secuencias crudas con FastQC (15) y de acuerdo a ello realizar su filtrado con Trimmomatic (16).

Tarea 3.2 Realizar el ensamblaje de las secuencias en *contigs* mediante dos software diferentes: MEGAHIT (17) y metaSPAdes (18).

Tarea 3.3 Comparar los resultados del ensamblaje utilizando el programa MetaQUAST (19).

Objetivo 4. Revisar las características de los métodos de *binning* y comparar el rendimiento de alguno de ellos.

Tarea 3.1 Revisar los programas de *binning* independientes de referencia.

Tarea 3.2 Realizar el *binning* de los *contigs* obtenidos por los dos ensambladores utilizando dos métodos diferentes: 1) Dependiente de abundancia: Canopy (8); y 2) Mixto: MetaBAT2 (20).

Tarea 3.3 Comparar los resultados del *binning* utilizando el programa CheckM (21).

Tarea 3.4 Analizar con qué programa(s) de ensamblaje y binning se obtienen los mejores resultados.

1.7.2 Calendario

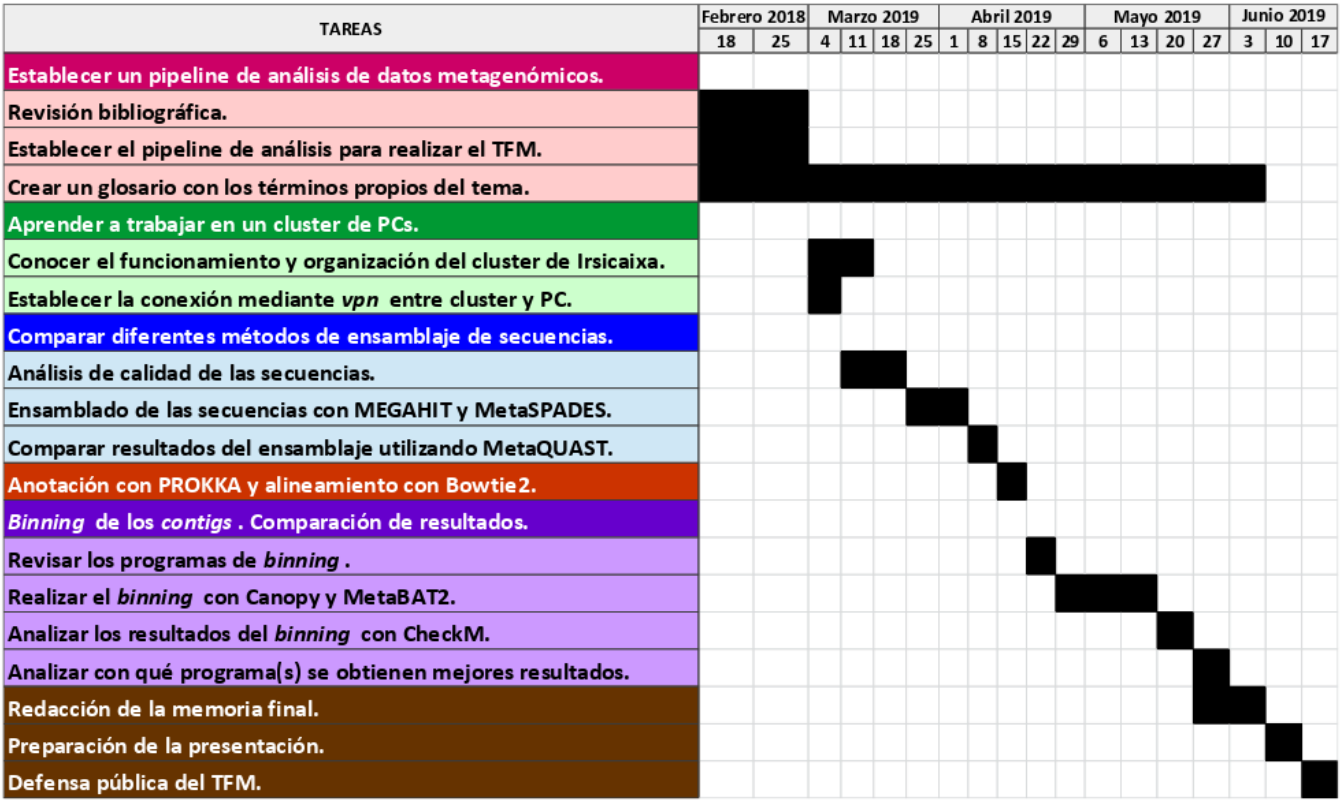

Figura 6. Diagrama de Gantt con las tareas planificadas.

1.7.3 Justificación de los cambios en caso necesario

Por temas logísticos, hemos empezado a trabajar con datos ya filtrados (*post-trimming)* según la calidad de las secuencias. Entre otros motivos, porque se requiere un paso previo de eliminación de ADN humano que queda fuera de los objetivos de este TFM. Sin embargo, el filtrado se ha llevado a cabo igualmente para poder conocer el software y trabajar en el *script* necesario. El análisis de calidad de las secuencias crudas, por tanto, no se ha realizado, pero sí el análisis de calidad de las secuencias tras el filtrado.

Por otro lado, se ha realizado la anotación de los genes a partir de cada uno de los ensamblajes y extraído las regiones codificantes de los contigs. Se han colapsado a un único fichero los catálogos single-samples. Las secuencias originales *post-trimming* se han alineado posteriormente frente al catálogo de genes generado. Es un paso previo al *binning*, aunque también podríamos haber hecho el *binning* mapeando las secuencias directamente sobre los *contigs* sin anotar.

En un principio estaba planificado comparar programas de *binning* de los tres tipos: basados en composición, basados en abundancia y mixtos. Sin embargo, tras realizar la revisión bibliográfica, se ha decidido comparar un método basado en abundancia (Canopy) y un método mixto (MetaBAT2). Se han excluido los métodos basados únicamente en composición porque presentan una serie de limitaciones que no los hacen óptimos para el análisis de datos metagenómicos: especies poco abundantes pueden no detectarse y para obtener resultados fiables necesitan secuencias más largas de lo que son nuestras secuencias de partida.

1.8 Relación de las actividades realizadas

1.8.1 Actividades previstas en el plan de trabajo

- Análisis de calidad de las secuencias y filtrado.
- Familiarización con los programas de ensamblaje y ensamblado de las secuencias tras el control de calidad.
- Comparación de los *contigs* obtenidos por ambos métodos.
- Elaboración de un glosario.
- Familiarización con los programas de *binning*: Canopy y MetaBAT2.
- Análisis del *binning* con CheckM.

1.8.2 Actividades no previstas y realizadas

- Unificación de los datos de calidad en un informe conjunto.
- Análisis del ensamblaje de una muestra con composición microbiana definida o *mock*.
- Anotación de los *contigs* y mapeo de las secuencias frente a los genes obtenidos.

2. Metodología

2.1 Análisis de calidad y *trimming*. FastQC y Trimmomatic

Los datos obtenidos del secuenciador o secuencias crudas no pueden ser analizados directamente. Primero, se analiza la calidad de estas secuencias crudas y, en función de lo que se observa, se realiza un filtrado para depurarlas y prepararlas para los análisis posteriores.

La calidad de las secuencias se ha analizado con el programa FastQC (15). En función de los datos obtenidos, se realiza un filtrado empleando el programa Trimmomatic (16) seleccionando los parámetros según la información que obtengamos con FastQC. Finalmente, se repetirá el análisis de calidad para las secuencias *post-trimmimng*. Los parámetros seleccionados para el *trimming* fueron los siguientes:

- **SLIDINGWINDOW:** Analizar la secuencia con una ventana de 4 bp, cortando cuando la calidad promedio por base cae por debajo de 20.
- **LEADING**: calidad mínima de las primeras bases de Q10.
- **TRAILING**: calidad mínima de las primeras bases de Q10.
- **MINLEN**: longitud mínima de las secuencias de 100 bp.

La información de todas las muestras por separados se integrará en único informe empleando el programa multiQC (22).

2.2 Ensamblaje. MEGAHIT Y MetaSPAdes

El ensamblaje es el proceso mediante el cual una gran cantidad de secuencias cortas de ADN se unen para crear una representación de los genomas originales a partir de los cuales se originaron las lecturas obtenidas. Existen dos opciones: alineamiento basado en una referencia y ensamblaje *de novo*. En el primer caso, un genoma previamente ensamblado se usa como referencia, sobre la que cada una de las secuencias se alinean en su posición más probable (Figura 7). Por el contrario, los ensamblajes del genoma *de novo* no asumen ningún conocimiento previo de la secuencia, el diseño o la composición de la secuencia del ADN fuente (4). No se utiliza ningún genoma como referencia, sino que se basa en el solapamiento de las distintas secuencias (Figura 7).

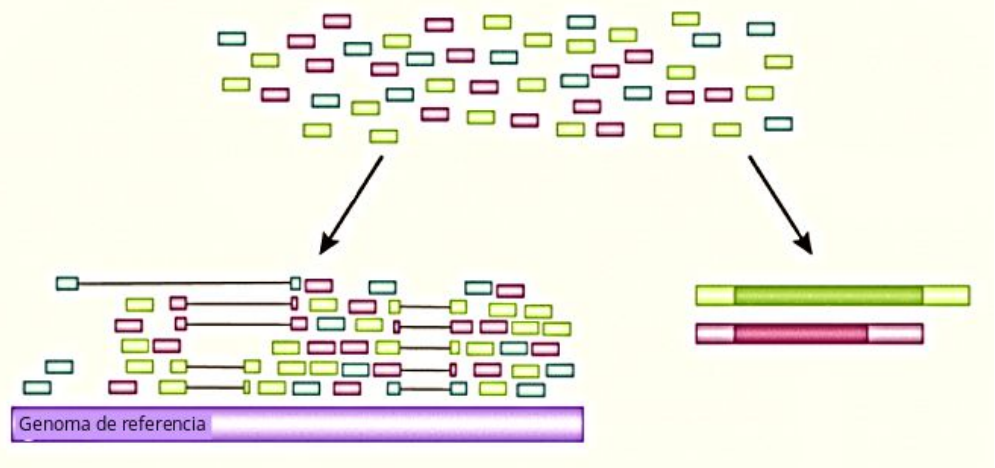

Figura 7. Tipos de ensamblaje o mapeo. Mapeo de lecturas a una referencia (izquierda) o ensamblaje *de novo* (derecha).

En este trabajo, para realizar el ensamblaje de las secuencias ya filtradas, se han empleado dos programa diferentes para el ensamblado *de novo* en su variante optimizada para secuencias metagenómicas : MEGAHIT (17) y metaSPAdes (18).

Las secuencias que van a usar los programas de ensamblaje son los archivos *fastq* de las secuencias ya filtradas en el directorio “Filtered”. En el caso de MEGAHIT, se han mantenido los parámetros definidos por defecto:

--min-count: multiplicidad mínima para filtrar (k_min + 1) -mers = 2

--k-min: tamaño mínimo de kmer = 21

--k-max: tamaño máximo de kmer = 99

--k-step: incremento en el tamaño del kmer para cada iteración = 10

En el caso de MetaSPAdes, no es necesario ajustar ningún parámetro, pero sí indicarle que son datos metagenómicos (--meta).

Los archivos resultantes de estos ensamblajes se guardarán en los directorios “MEGAHIT_Assembly” y “METASPADES_Assembly”, respectivamente.

2.3 Evaluación del ensamblaje. QUAST / MetaQUAST

MetaQUAST (19) es una adaptación del programa QUAST (24) para poder realizar la evaluación de los ensamblajes de datos metagenómicos, en lugar de hacerlo sobre genomas únicos. Detecta errores basándose en alineamientos con secuencias de referencia y proporciona datos estadísticos sobre los *contigs*. Uno de los datos que proporciona es el N50, parámetro clave para analizar la calidad de un ensamblaje. Para calcularlo se siguen los siguientes pasos:

1. Se genera un ranking de todos los *contigs* en base a su tamaño, de más grande a más pequeño.

2. Se suman las bases que contienen, desde el *contig* más grande al más pequeño.

3. El N50 corresponde al tamaño del *contig* situado en el ranking donde ya llevas acumuladas el 50% de todas las bases del genoma.

El L50 es otro parámetro que también se emplea para evaluar la calidad de un ensamblaje y es el menor número de *contigs* cuya longitud total representa al menos la mitad de la longitud del ensamblaje.

MetaQUAST puede ser utilizado usando dos modos: “multiple reference” y “de novo detection”. En el primer caso, el usuario dispone de una idea previa sobre el contenido de la muestra metagenómica que ha generado el ensamblaje y propone un conjunto de genomas bacterianos pasados como parámetro, mientras que en el segundo caso se le pide al programa que identifique las especies bacterianas mayoritarias en los *contigs* por homología de secuencia y las descargue en tiempo real desde una base de datos externa (NCBI, por defecto).

El pipeline con múltiples referencias consta de cuatro pasos principales (Figura 5):

1. Todos los genomas de referencia se concatenan en un archivo (referencia combinada). QUAST se alimenta con todos los ensamblajes de entrada frente a la referencia combinada.

2. Los *contigs* se dividen en grupos, cada uno de los cuales contiene secuencias mapeadas frente a un genoma de referencia concreto. Los *contigs* asignados a varios genomas forman parte de varios grupos. Los *contigs* no alineados van en un grupo adicional.

3. QUAST se lanza para cada referencia por separado, alimentándose con un grupo correspondiente de *contigs*. El grupo de *contigs* no alineados es procesado sin ninguna referencia de entrada.

4. Finalmente, todos los resultados de QUAST se agrupan en una serie de informes y visualizaciones.

Para realizar el análisis con MetaQUAST, utilizamos los archivos “final.contigs.fa” y “contigs.fasta” obtenidos para cada muestra tras realizar el ensamblaje con MEGAHIT y metaSPAdes, respectivamente. Además de las muestras incluidas en el estudio, se ha secuenciado una muestra con composición de bacterias conocida, también denominada *mock*, que nos permite comprobar si la secuenciación y el resto de pasos hasta ahora han ido bien.

En el caso de la *mock*, conocemos la composición bacteriana de la muestra. Así pues, para evaluar el ensamblado vamos a utilizar el listado de cepas de referencia correspondiente (anexo 6.3) (*reference based evaluation*). Sin embargo,

en el caso de las muestras no vamos a proporcionar secuencias de referencia a MetaQUAST (Figura 8), sino que en función de cada muestras las buscará en NCBI (*de novo evaluation*).

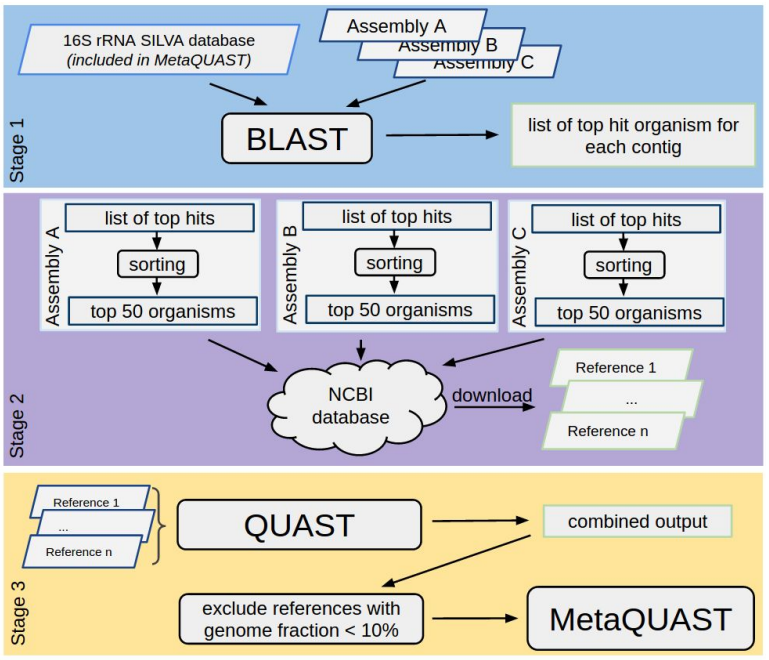

Figura 8. Pipeline de MetaQUAST para la evaluación de ensamblajes *de novo* (sin proporcionar referencias). Paso 1: Alineamiento de los ensamblajes a las secuencias 16S rRNA de la base de datos SILVA (25) mediante BLASTn (26). Para cada *contig*, se elige el organismo con el máximo score de BLAST. Paso 2: los 50 microorganismos con las puntuaciones máximas de BLAST son escogidos. MetaQUAST intenta encontrar y descargar los genomas de referencia para todos estos organismos desde la base de datos NCBI. Paso 3: todos los archivos descargados se concatenan en un solo archivo de referencia. QUAST utiliza este archivo para estimar las coberturas para cada referencia y ensamblaje. MetaQUAST excluye los genomas de referencia con una cobertura baja (menos del 10%) y lanza

el pipeline utilizando los archivos restantes. *Fuente: Mikheenko A, Saveliev V, Gurevich A. MetaQUAST: evaluation of metagenome assemblies. Bioinformatics 2016;32(7):1088-90.*

Puesto que este paso es muy costoso computacionalmente, ya que para cada muestra se descargan sus correspondientes genomas de referencia, y además había problemas con la conexión a Internet del clúster, se ha optado por utilizar *quast.py* (QUAST). En este caso, no se utilizan secuencias de referencias y, por lo tanto, no las tiene que descargar desde NCBI, evitando los problemas mencionados y siendo además mucho más rápido. La diferencia es que usando *quast.py* no se va a obtener la información sobre *missassemblies and structural variations*, *Genome representation and its functional elements*, ni *Variations of N50 based on aligned blocks*. No obstante, se ha realizado el análisis con *metaquast.py* para una muestra y así poder analizar los resultados por este método.

A partir de los resultados obtenidos, se extraerán los datos y se construirá una tabla que nos permitirá comparar los dos programas de ensamblado con el programa R (versión 3.4.0) y decidir si una de las opciones es mejor que la otra.

2.4 Anotación. PROKKA

La anotación del genoma es el proceso que consiste en identificar y etiquetar todas las características relevantes en una secuencia del genoma. Como mínimo, esto debe incluir las coordenadas de las regiones de codificación predichas y sus productos putativos. Prokka (27) es un software de anotación de genomas bacterianos basado en línea de comando. Podríamos prescindir de la anotación, pero nos permite comprobar qué diversidad génica es capaz de capturar cada método.

A partir de este punto, los *contigs* de menos de 1000 bp se descartan para así filtrar *contigs* con una menor fiabilidad. Los *contigs* restantes son el input para hacer la anotación con el programa PROKKA. Los mismos pasos se realizarán tanto para los *contigs* obtenidos con MEGAHIT como con y MetaSPAdes.

2.5 Alineamiento o *mapping*. Bowtie 2

Hasta el momento solo tenemos el número de genes y anotaciones en la muestra. Debido a que estas anotaciones se predicen a partir de ensamblajes, hemos perdido la información cuantitativa. Para poder cuantificar los genes, necesitamos mapear las lecturas al catálogo de genes obtenido en la anotación.

Bowtie2 (28,29) es una herramienta muy rápida y eficiente computacionalmente para alinear secuencias a genomas/genes de referencia. Bowtie2 indexa el genoma basándose en la transformación de Burrows–Wheeler

(BWT del inglés Burrows–Wheeler transform, también conocida como compresión por ordenación de bloques). Al igual que un índice para un libro, la creación de un índice para una base de datos genómica permite un acceso rápido a cualquier "registro". En el caso de los programas de mapeo, la creación de un índice para una secuencia de referencia le permite ubicar más rápidamente una lectura en esa secuencia en una ubicación en la que sabe que al menos una parte de la lectura coincide a la perfección o solo con algunas discrepancias. Al saltar directamente a estos puntos del genoma, en lugar de intentar alinear completamente la lectura con cada lugar del genoma, se ahorra mucho tiempo.

En general, el primer paso consiste en indexar el archivo de referencia, independientemente del programa de alineamiento que se utilice. Para conseguir este archivo de referencia, hay que generar un archivo *fasta* con todos los genes anotados previamente en todas las muestras, colapsando aquellos genes duplicados o, dicho de otra manera, aquellos genes que se hayan obtenido con la misma secuencia exacta en dos muestras o más, y quedándose con una única copia de los mismos. Para ello, se han utilizado las funciones *fasta_formatter* y *fastx_collapser* del paquete FASTX-Toolkit (30).

El comando *bowtie2-build* construye un index de Bowtie a partir de un conjunto de secuencias de ADN. Este índice permite hacer búsquedas extremadamente rápidas en las secuencias del archivo de referencia que, de otra forma, serían sumamente costosas computacionalmente. Genera un conjunto de 6 archivos con los sufijos *.1.bt2*, *.2.bt2*, *.3.bt2*, *.4.bt2*, *.rev.1.bt2* y *.rev.2.bt2*. En el caso de index grandes, estos sufijos tendrán una terminación *bt2l*. Todos estos archivos

conjuntamente constituyen el index, necesario para alinear las lecturas con esa referencia. Bowtie 2 ya no usa la secuencia original de los archivos FASTA una vez que se construye el index. Solo se necesita indexar los genes de referencia una vez, independientemente de cuántas muestras se vayan a mapear, ya que hemos creado un archivo multifasta con todos los genes únicos obtenidos de los ensamblajes de todas las muestras.

El siguiente paso consiste en mapear las secuencias. Para ello, el comando que se utiliza es *bowtie2*.

Los mismos pasos se realizarán para las anotaciones de los *contigs* obtenidos con MEGAHIT y MetaSPAdes.

2.6 *Binning*. Canopy y MetaBAT2

A partir de los genes o *contigs*, las secuencias se van a agrupan basándose en su composición y/o abundancia mediante diferentes algoritmos sin necesidad de una referencia. Los métodos de *binning* suponen que la abundancia de genes de una misma especie covaría para un mismo taxón y/o que los *contigs* de un mismo *bin* tienen frecuencias de uso de k-meros (todas las posibles combinaciones de nucleótidos de longitud k que están contenidas en una secuencia) similares. De esta manera, los contigs se agrupan en *bins*. Los *reads* dentro de estos *bins* se pueden volver a ensamblar de nuevo, generando contigs de mejor calidad (menos y más largos) y cada grupo de secuencias se corresponde con una especie metagenómica diferente, que asumimos que son consistentes.

Existen los siguientes tipos de métodos de binning: basados en la composición de k-meros de las secuencias; basados en la co-abundancia de secuencias entre muestras; y mixtos, que tienen en cuenta tanto la composición como la abundancia.

El punto de partida para este paso son los archivos BAM. En primer lugar, debemos generar una tabla que recoja cuántos *reads* tenemos para cada gen en cada muestra (tabla de abundancias). Un vez tengamos esta tabla, podremos hacer el *binning* con Canopy (8) y MetaBAT2 (13,20). Estos pasos se repetirán tanto para los ensamblajes realizados con MEGAHIT como con MetaSPAdes.

El método de *binning* Canopy se basa en agrupar las entidades biológicas o especies metagenómicas según la co-abundancia de genes, generando los CAG o *co-abundant groups of genes* (Figura 9). Antes de poder usar el programa, se debe generar la tabla de abundancias. A partir de esta tabla y el catálogo de genes, ya se puedo realizar el *binning*. Entre los parámetro a controlar, destacan los siguientes:

-n [--num_threads] (=4): número de cpu a usar.

--max_canopy_dist (=0.1): máxima diferencia de correlación Pearson entre el centro del *canopy* y un punto incluido en el *canopy*.

--max_merge_dist (=0.1): máxima diferencia de correlación Pearson entre los centros de dos *canopies* en los que los *canopies* debería coincidir.
--profile_measure (=75Q): medida de abundancia de genes.

--filter_min_obs (=3): descarta aquellos perfiles que tienen menos de N muestras.

--filter_max_top3_sample_contribution (=0.9): descarta aquellos perfiles para los que las tres muestras en las que son más frecuentes suponen más de X% de la señal.

--cag_filter_min_sample_obs (=3): devolver solo aquellos *canopies* que tienen al menos N clústers no vacíos.

--cag_filter_max_top3_sample_contribution (=0.9): descartar *canopies* donde tres o más muestras representan más del x% de la señal.

--stop_criteria (=50000): detener el *clustering* tras X ciclos.

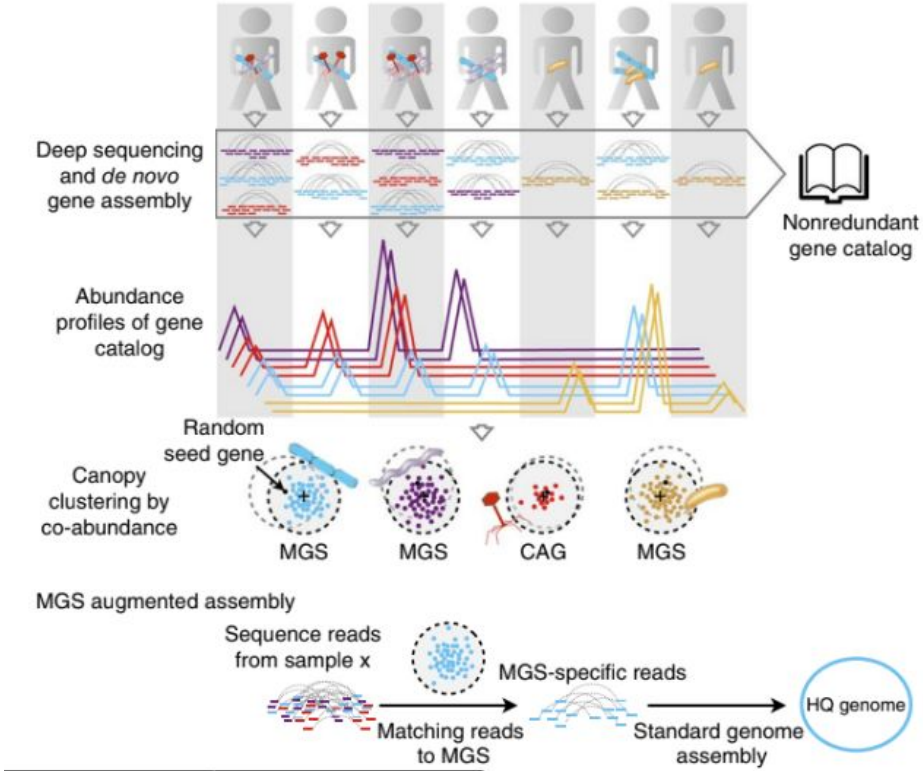

Figura 9. Esquema del flujo de trabajo de Canopy. *Fuente: Nielsen HB, Almeida M, Juncker AS, Rasmussen S, Li J, Sunagawa S, et al. Identification and assembly of genomes and genetic elements in complex metagenomic samples without using reference genomes. Nature Biotechnol 2014;32:822-8.*

Después del *clustering* con Canopy, los *canopies* deben filtrarse: *canopies* raros, demasiado pequeños o *outliers*.

- Filtrar genes *outliers*. Todos los genes deben tener un coeficiente de correlación de spearman superior a 0.7. Este filtro está destinado a eliminar genes de los CAG.
- Filtrar *canopies outliers*. El 90% del perfil total de *canopies* deben originarse a partir de más de tres muestras. Este filtro elimina los CAG completos que son controlados por valores atípicos, así como los GAC basados en muy pocas observaciones.
- Filtrar canopies demasiado pequeños. Todos los CAG con menos de tres genes deben excluirse. Se analizará sólo para el ensamblador MetaSPAdes.

En el caso de MetaBAT2, los *bins* se forman a partir no sólo de la abundancia de genes sino de la frecuencias de k-meros (Figura 10). En este caso, sólo se necesitan los archivos BAM y el catálogo de genes, ya que el mismo programa se encarga de generar la tabla de abundancia y la frecuencia de k-meros.

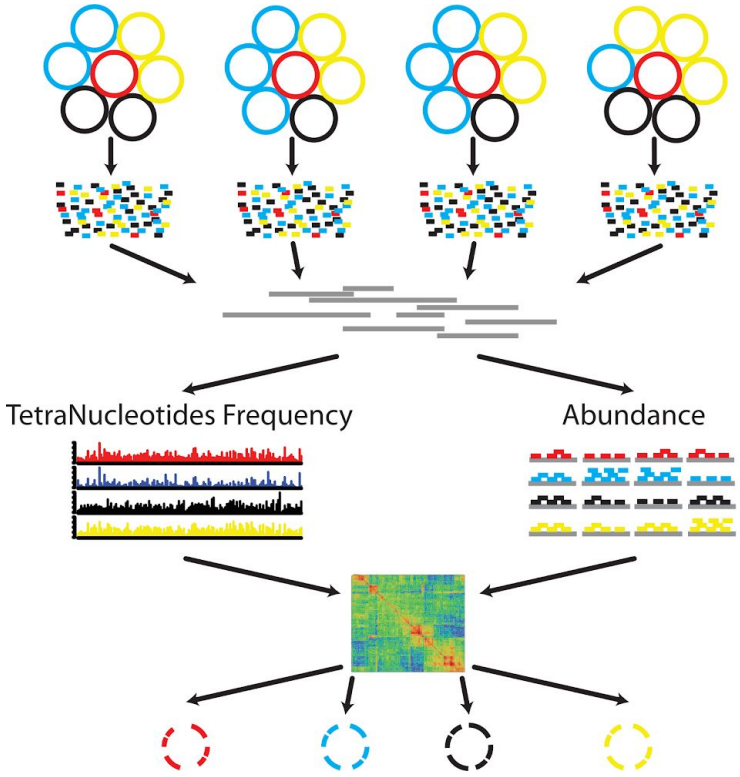

Figura 10. Esquema del flujo de trabajo con MetaBAT2. *Fuente: Kang DD, Froula J, Egan R, Wang Z. MetaBAT, an efficient tool for accurately reconstructing single genomes from complex microbial communities. PeerJ 2015;3:e1165.*

2.7 Evaluación del *binning*. CheckM

Las preguntas que debemos formular ahora son: ¿debería estar este gen realmente en este *bin*? ¿pertenecen todos los genes de un mismo *bin* a un mismo ente microbiológico? Desafortunadamente, analizar la calidad del *binning* no es tan sencillo como lo era analizar la de los ensamblajes. En los estudios de metagenómica, se emplea la presencia/ausencia de genes marcadores de copia única para evaluar cómo de completo está un genoma (*completeness*). Este método presenta dos limitaciones: la distribución de los genes marcadores a lo largo del

genoma es desigual y se encuentran en baja cantidad (típicamente menos del 10% de todos los genes). Estas limitaciones se han solucionado parcialmente mediante la identificación de genes que son ubicuos y de copia única dentro de un *phylum* específico, lo que aumenta el número de genes marcadores utilizados en la estimación. Además, podemos estimar la posible contaminación si nos encontramos genes marcadores de copia única varias veces dentro de un genoma.

CheckM (21) es un método automatizado para estimar la *completeness* y contaminación de un genoma utilizando genes marcadores de copia única que son específicos del linaje inferido de un genoma dentro de un *reference genome tree*. La *completeness* se estima a partir del número de genes de copia única (SCGs, *single copy genes*) presentes en el *bin*. Por su parte, la contaminación se estima calculando cuántos SCGs hay en múltiples copias, ya que solamente debería haber una copia por SCG por genoma.

El input para CheckM es el directorio que contiene los *bins* a analizar en formato FASTA. CheckM consiste en una serie de comandos que permite varios análisis y flujos de trabajo (*workflow*) diferentes. Se va a utilizar el *workflow* lineage_wf, que incluye distintos análisis:

- **tree**: coloca los *bins* en el *reference genome tree*
- **lineage_set**: infiere grupos de marcadores específicos de cada linaje en cada *bin*
- **analyze**: identifica genes marcadores en los *bins*
- **qa**: analiza la contaminación y *completeness* de los *bins*

2.8 Visualización y análisis. MEGAN

Como las técnicas de *binning* realizan una agrupación de las secuencias de una forma no supervisada, el uso de herramientas para la visualización de los *bins* es muy recomendable. Una de las posibles opciones es usar MEGAN (32).

En primer lugar, se realiza un BLAST (26) de las secuencias de todos *bins* obteniendo un formato adecuado para poder usado en MEGAN. En este caso, se ha optado por el formato -m 7, que se corresponde con el formato XML. Este paso consiste en un alineamiento local de las secuencias dentro de cada *bin* frente a la base de datos “nt” de NCBI (secuencias nucleotídicas parcialmente no redundantes). Se utiliza “blastn” ya que se está comparando una secuencia de nucleótidos contra una base de datos que contiene también secuencias nucleotídicas. El objetivo es encontrar a qué especies pertenecen los genes contenidos en cada *bin* para así asignar la taxonomía y ver la composición de cada *bin*. De esta forma, podremos saber si un *bin* contiene genes de una única especie o no.

El siguiente paso consiste en transformar estos ficheros obtenidos tras realizar el BLAST, en un formato adecuado para que MEGAN pueda interpretarlo: formato *rma*. Una vez obtenidos estos ficheros, se realiza una comparación de los distintos *bins*, obteniéndose diferentes visualizaciones de los mismos. La versión empleada es MEGAN6.

2.9 Elaboración de figuras.

Las figuras propias se han elaborado con los programas yEd Graph Editor (https://www.yworks.com/products/yed) e Inkscape (https://inkscape.org/es/).

2.10 Análisis estadísticos.

Para el análisis estadístico se ha empleado R (v 3.4.0) y RStudio (v 1.0.153).

3. Resultados

3.1 Análisis de calidad y *trimming*. FastQC y Trimmomatic

En la figura 11, se muestra la calidad de las secuencias de una de las muestras tal como se obtiene con el programa FastQC. Se adjunta también como anexo 6.2 el informe conjunto de todas las muestras. La mediana (rango intercuartílico, RIQ) del número de lecturas ya filtradas por muestra es de 14,3 (12,0-17,6) millones. El rendimiento de la secuenciación no es igual para todas las muestras.

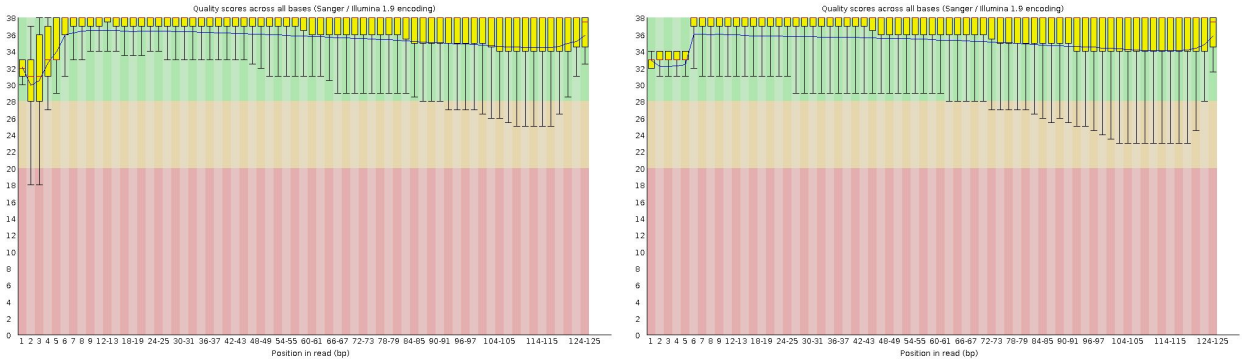

Figura 11. Calidad de las secuencias en la misma muestra después del filtrado con el software Trimmomatic. A la izquierda se muestran las lecturas *forward* y a la derecha las *reverse*.

3.2 Ensamblaje. MEGAHIT Y MetaSPAdes

En ambos casos, se obtiene un directorio para cada muestra. Dentro de este directorio, el archivo que contiene los *contigs* es "final.contigs.fa" en el caso de MEGAHIT y "contigs.fasta" en el caso de MetaSPAdes. Estos contigs serán los que usemos para realizar el *binning*, pero antes de este paso, vamos a analizar la calidad del ensamblaje.

3.3 Evaluación del ensamblaje. QUAST / MetaQUAST

Con los datos contenidos mediante QUAST, se observa que para estos datos los resultados del ensamblaje son mejores para MetaSPAdes que MEGAHIT. El N50 es muy superior para MetaSPAdes que para MEGAHIT. Además, MetaSPAdes generó un número ligeramente superior de contigs con una longitud superior a los 1000 pares de bases, una longitud total del ensamblaje superior, así como el *contig* más largo (Tabla 1, Figura 12).

	MEGAHIT	**metaSPAdes**	***p*-valor**
N contigs	**75.874** (58.111-89.237)	**77.057** (57.110-93.065)	**0,02**
N contigs >= 1000 bp	**3.1591** (26.261-37.979)	**32.386** (25.359-38.160)	**0,005**
Longitud total	**134.830.862** (111.489.872-163.001.066)	**151.835.181** (123.096.516-177.861.144)	**<0,001**
Longitud total contigs >= 1000 bp	**103.196.977** (86.947.270-126.694.311)	**119.429.664** (98.428.760-143.733.194)	**<0,001**
Contig más largo	**248.455** (193.031-324.921)	**385.269** (303.424-535.078)	**<0,001**
N50	**2.834** (2.424-3.788)	**3.723** (3.030-5.020)	**<0,001**
L50	**8.175** (5.226-10.962)	**6.696** (4.144-9.263)	**<0,001**

Tabla 1. Resumen del análisis de los ensamblajes realizados sobre las mismas muestras con MEGAHIT y MetaSPAdes. Se indica la mediana (rango intercuartílico) de los principales parámetros obtenidos por QUAST y el valor *p* calculado mediante un test de Wilcoxon para muestras pareadas.

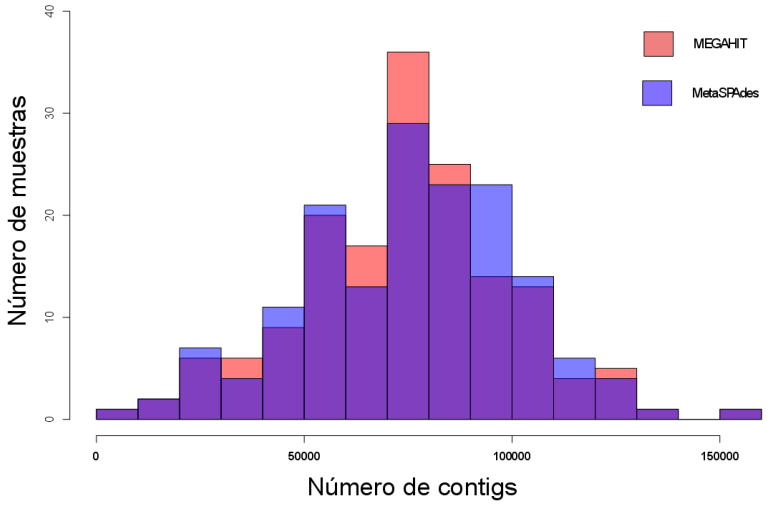

Figura 12. Distribución del número de *contigs* según el ensamblador empleado.

Cuando se analiza el *mock,* usando un conjunto de referencias conocidas, vemos que los resultados son también mejores cuando se utiliza MetaSPAdes (Tabla 2, Figura 13). Hay algunas especies de la *mock* para las que la cobertura del genoma con el ensamblaje es muy buena, pero tenemos otros para los que es muy baja. Al tratarse de una *staggered mock* (la abundancia de cada uno de los microorganismos en la *mock* varía según la especie), podría ser que aquellas bacterias menos cubiertas por el ensamblaje sean las menos abundantes en la *mock* original. También podría estar relacionado con el tamaño del genoma, aunque

no se observa una relación lineal entre estos dos parámetros (Figura 14). Cabe remarcar, aún así, que las medidas de fracción genómica cubierta son coherentes entre los dos ensambladores. Resultados similares se observan al analizar una muestra real (*Sample1*) usando *metaquast.py* (Tabla 3).

	MEGAHIT	**metaSPAdes**
Fracción genoma cubierta (%)	**42,81**	**45,724**
Acinetobacter_baumannii	**27,798**	**40,962**
Bacillus_cereus	**14,148**	**13,814**
Bacteroides_vulgatus	**0,06**	**0,111**
Clostridium_beijerinckii	**72,369**	**74,619**
Deinococcus_radiodurans	**0,056**	**0,1**
Enterococcus_faecalis	**0,055**	**0,279**
Helicobacter_pylori	**81,139**	**81,792**
Lactobacillus_gasseri	**14,564**	**18,601**
Listeria_monocytogenes	**24,729**	**39,198**
Neisseria_meningitidis	**22,002**	**31,077**
Propionibacterium_acnes	**29,541**	**47,84**
Pseudomonas_aeruginosa	**91,227**	**91,253**
Rhodobacter_sphaeroides	**87,55**	**87,734**
Schaalia_odontolytica	**-**	**0,026**
Staphylococcus_aureus	**95,115**	**94,508**
Staphylococcus_epidermidis	**84,684**	**84,979**
Streptococcus_agalactiae	**89,192**	**89,055**
Streptococcus_mutans	**0,322**	**0,342**
Streptococcus_pneumoniae	**0,143**	**0,294**
Alineamiento más largo	**194.401**	**253.411**
Longitud total del alineamiento	**26.879.968**	**28.700.138**
Mismatches	**431.726**	**381.707**
N contigs	**8.893**	**9.417**
N contigs >= 1000 bp	**4.171**	**4.055**
N contigs >= 5000 bp	**1.334**	**999**
N contigs >= 10000 bp	**802**	**641**
N contigs >= 25000 bp	**358**	**336**

N contigs >= 50000 bp	**161**	**195**
Contig más largo	**356.789**	**429.886**
Longitud total	**41.674.017**	**43.898.801**
N50	**28.228**	**53.709**
L50	**320**	**186**

Tabla 2. Principales parámetros de calidad obtenidos por QUAST de los ensamblajes realizados sobre la muestra *mock* usando MEGAHIT y MetaSPAdes. La fracción del genoma cubierta corresponde con el número total de bases alineadas en las referencias (suma de las bases ensambladas), dividido por el total de bases que componen el total de genomas de referencia (suma de los tamaños de los genomas de referencia). Además, se muestra este valor de cobertura para cada una de las referencias por separado.

	MEGAHIT	**metaSPAdes**
Fracción genoma cubierta (%)	**53,379**	**57,339**
Alistipes_finegoldii_DSM_17242	**10,252**	**14,585**
Alistipes_putredinis_DSM_17216	**76,781**	**77,45**
Alistipes_shahii	**68,008**	**72,871**
Bacteroides_uniformis	**58,229**	**68,777**
Bacteroides_uniformis_dnLKV2	**64,463**	**64,888**
Bacteroides_uniformis_str._3978_T3_ii	**52,141**	**50,934**
Bacteroides_vulgatus	**73,61**	**75,378**
Bacteroides_vulgatus_ATCC_8482	**63,901**	**63,049**
Barnesiella_intestinihominis_YIT_11860	**90,168**	**90,464**
Butyrivibrio_crossotus	**79,399**	**79,651**
Candidatus_Alistipes_marseilloanorexicus_AP11	**7,186**	**15,165**
Dialister_invisus	**83,112**	**83,614**
Dorea_formicigenerans_ATCC_27755	**22,249**	**30,499**
Dorea_longicatena	**49,037**	**59,33**
Faecalibacterium_prausnitzii	**8,903**	**9,645**
Gemmiger_formicilis	**9,586**	**14,105**
Odoribacter_splanchnicus_DSM_20712	**27,269**	**39,744**

Parabacteroides_distasonis	**68,732**	**75,63**
Parabacteroides_merdae	**56,807**	**67,932**
Paraprevotella_clara	**62,029**	**66,709**
Ruminococcus_sp._5_1_39BFAA	**22,941**	**30,425**
Ruminococcus_torques_ATCC_27756	**48,46**	**61,208**
Alineamiento más largo	**228.805**	**202.757**
Longitud total del alineamiento	**37.178.659**	**41.157.923**
Mismatches	**937**	**856**
N contigs	**68.898**	**74.210**
N contigs >= 1000 bp	**28.383**	**30.914**
N contigs >= 5000 bp	**3.079**	**3.447**
N contigs >= 10000 bp	**1.242**	**1.386**
N contigs >= 25000 bp	**328**	**444**
N contigs >= 50000 bp	**96**	**171**
Contig más largo	**300.298**	**353.900**
Longitud total	**116.578.447**	**134.349.693**
N50	**2.463**	**2.906**
L50	**8.000**	**7.238**

Tabla 3. Principales parámetros de calidad obtenidos por METAQUAST de los ensamblajes realizados sobre una muestra real usando MEGAHIT y MetaSPAdes. Se incluyen las referencias seleccionadas por METAQUAST.

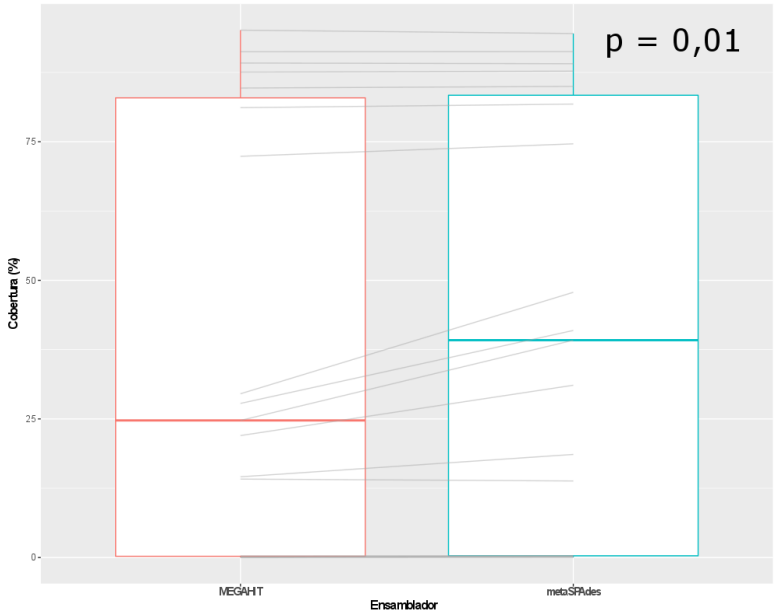

Figura 13. Representación de las coberturas obtenidas por cada ensamblador para cada microorganismo de la *mock* y valor *p* calculado mediante un wilcoxon-test de muestras pareadas.

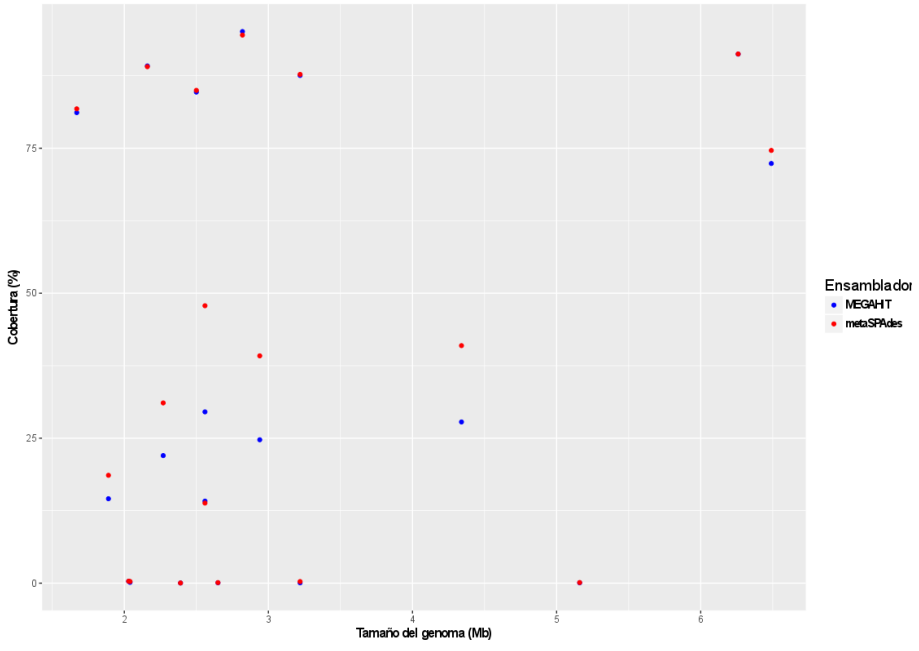

Figura 14. Correlación entre el tamaño del genoma y la cobertura de cada uno de ellos (MEGAHIT a la izquierda y MetaSPAdes a la derecha).

Por otro lado, en las muestras reales hay que considerar que la bondad del ensamblaje dependerá tanto del número de secuencias que hayamos obtenido de cada una de las muestras como de la diversidad bacteriana que albergan esas muestras. Por lo tanto, ya que no todas las muestras parten del mismo número de secuencias, los resultados del ensamblaje pueden variar entre muestras dentro de una misma estrategia, ya que hemos usado todas las secuencias de cada muestra para el ensamblaje. Se ha optado por esta estrategia en lugar de escoger un *subset* para no perder información de secuencias, ya que la evaluación del ensamblaje no es el objetivo primario de este trabajo. Al analizar la correlación entre el número de secuencias por muestra y el valor de N50 para cada ensamblaje sólo se observa una ligera correlación positiva (r=0,27 y r=0,32 para MEGAHIT y metaSPAdes, respectivamente) (Figura 15). Al final, además, se calculará la abundancia relativa de cada taxón por muestra, es decir, lo estaremos normalizando por el número de secuencias totales.

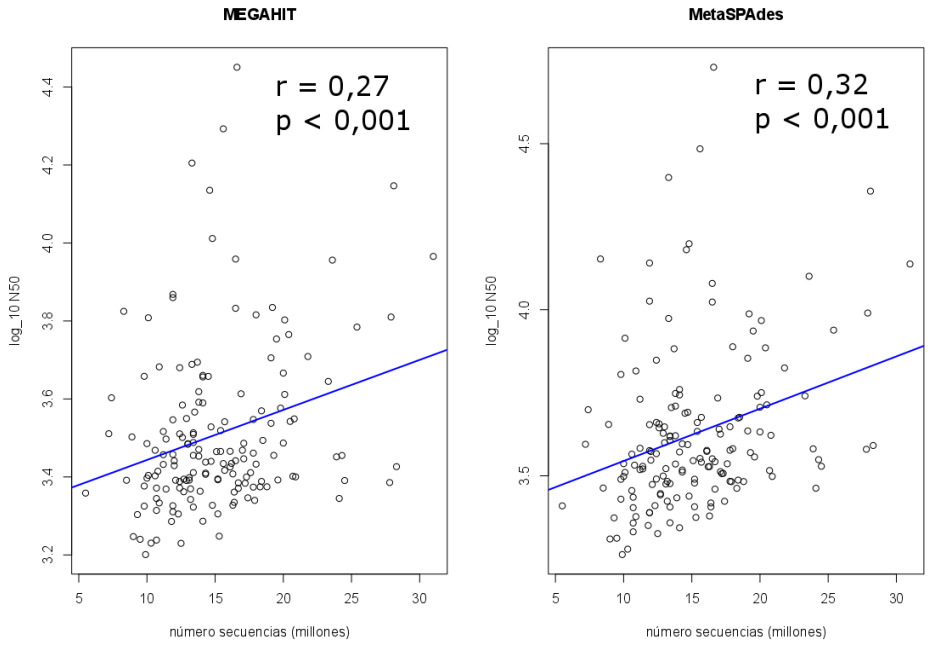

Figura 15. Correlación entre el número de secuencias/muestra iniciales usado por el ensamblador (MEGAHIT a la izquierda y MetaSPAdes a la derecha) y el N50 obtenido con QUAST para cada muestra.

3.4 Anotación. PROKKA

Se obtiene un directorio para cada muestra y dentro de este directorio un listado de los diferentes genes anotados.

3.5 Alineamiento o *mapping*. Bowtie 2

La generación del index ha tardado unos 15 minutos y ha generado los siguientes archivos:

```
AllCommonGenesINDEX.1.bt2  AllCommonGenesINDEX.3.bt2
AllCommonGenesINDEX.rev.1.bt2
AllCommonGenesINDEX.2.bt2  AllCommonGenesINDEX.4.bt2
AllCommonGenesINDEX.rev.2.bt2
```

El mapeo se ha realizado en paralelo. El archivo de salida inicial está en formato SAM (Sequence Alignment/Map format). Es un formato de texto delimitado por tabulador que consiste en un encabezado opcional y una sección de alineación. El encabezado comienza con '@', mientras que las líneas de alineación no. Este archivo contiene toda la información sobre dónde mapea cada lectura en la referencia, la calidad de este mapeo y los cambios de nucleótidos respecto a la

referencia, incluyendo inserciones y deleciones, entre otros. Como este archivo ocupa mucho espacio y además no es el formato que utilizarán posteriormente los programa de *binning*, los transformamos en el mismo paso a formato *sorted* BAM, mediante los comandos *samtools view* y *samtools sort* (SAMTOOLS) (15), ordenándose por coordenadas. En definitiva, conseguimos que este paso sea más rápido y que los archivos generados ocupen menos espacio en el disco, siempre persiguiendo un uso óptimo de los recursos. Hay que tener en cuenta que, cuando se alinean los archivos FASTQ, las alineaciones producidas están en orden aleatorio. Es por esta razón que debemos ordenar los archivos BAM según las coordenadas en el genoma antes de poder utilizarlos. En nuestro caso, ese orden es arbitrario, ya que no tenemos un genoma como referencia sino una colección de genes. Estos archivos *sorted* BAM serán el input para el proceso de *binning*. Como ya se ha comentado, el input para el *binning* lo podríamos haber conseguido también directamente a partir de los *contigs*, sin necesidad de hacer la anotación.

3.6 *Binning*. Canopy y MetaBAT2

Con MetaBAT2 hemos obtenido ocho *bins* para MEGAHIT y siete para MetaSPAdes, mientras que con canopy se han obtenido 3412 para MEGAHIT y 2562 para MetaSPAdes. Tras el filtrado, quedan 19 *bins* que contienen más de 700 genes. El procesamiento con MetaBAT2 ha sido bastante más sencillo y el tiempo de procesamiento más corto. Sin embargo, vemos que el número de *bins* es menor que para Canopy.

3.7 Evaluación del *binning*. CheckM

Se adjuntan como anexos los resultados para cada ensamblador y método de *binning*. Con MetaBAT2 la *completeness* no ha superado el 5% para ninguno de los *bins*. En el caso de canopy, se obtienen 34 *bins* con una *completeness* superior al 50%. Sin embargo, es necesaria la visualización de los mismos para completar la evaluación de su calidad.

3.8 Visualización y análisis. MEGAN

Se han seleccionado los *bins* generados a partir del ensamblaje con MetaSPAdes, porque ya se observó que el ensamblaje con este programa era mejor que el obtenido con MEGAHIT. En el caso de Canopy, se han incluido los *bins* con una *completeness* igual o superior al 50%. Como se puede observar en la Figura 16, aunque los primeros *bins* son de buena calidad, hay *bins* que se corresponden con la misma especie y otros muchos *bins* contienen genes de especies diferentes. En el caso de MetaBAT2, los siete *bins* incluyen genes de diferentes especies (Figura 17).

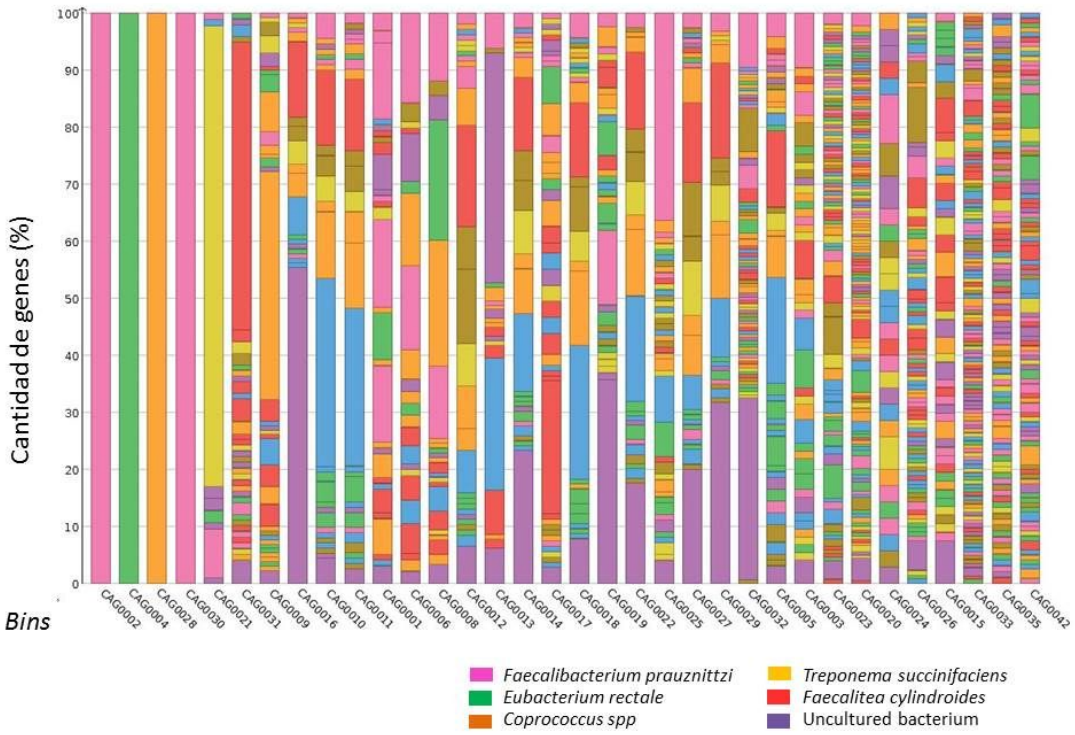

Figura 16. Representación de los *bins* obtenidos tras el ensamblaje con MetaSPAdes y el *binning* con Canopy. Cada barra representa un *bin* o especie metagenómica y en el eje vertical se representa el porcentaje de genes. Cada color representa una especie diferente. En la leyenda se muestran sólo algunas de las especies.

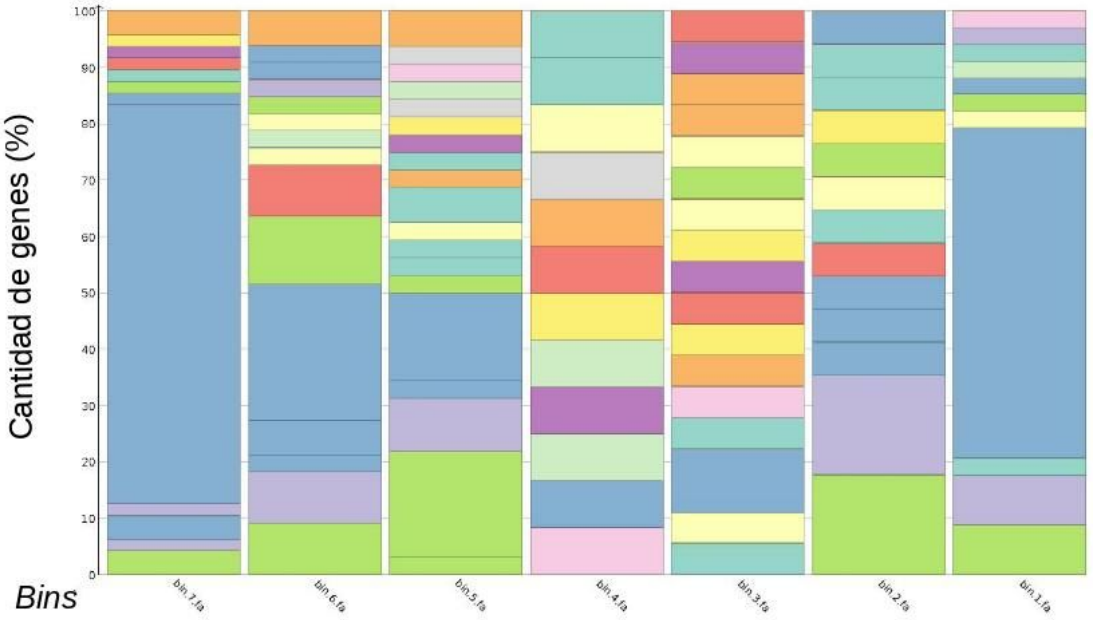

Figura 17. Representación de los *bins* obtenidos tras el ensamblaje con MetaSPAdes y el *binning* con MetaBAT2. Cada barra representa un *bin* o especie metagenómica y en el eje vertical se representa el porcentaje de genes.

4. Discusión

En relación al ensamblaje, se han obtenido mejores resultados utilizando MetaSPAdes que MEGAHIT. Esto concuerda con estudios previos (33,34). Como ha quedado reflejado, la cobertura conseguida con el ensamblaje parece no estar influenciada por el tamaño del genoma ni por el número de secuencias de las muestras. Con el objetivo de profundizar un poco más en el tema, se ha solicitado a la casa comercial la información sobre la abundancia de cada una de las especies en la *mock* para comprobar también si la cobertura conseguida para cada especie está relacionada con la abundancia de cada una de ellas.

Respecto al *binning*, Canopy requiere una manipulación previa de los datos para obtener la tabla de abundancias, mientras que MetaBAT2 admite los archivos BAM y el catálogo de genes o los *contigs* directamente. Sin embargo, los resultados obtenidos indican que Canopy ha generado muchos más *bins* que MetaBAT2. Podría parecer que el tener en cuenta la composición de las secuencias además de la co-abundancia de genes está teniendo un efecto negativo en este caso. Sin embargo, al visualizar los *bins* obtenidos, se ha comprobado que los resultados tampoco son buenos para Canopy. Aunque se han obtenido algunos *bins* que corresponden a especies concretas, se han obtenido *bins* que corresponden a la misma especie y otros que contienen genes de especies diferentes. Cabe remarcar que algunos *bins* se corresponden con especies no cultivadas y que sería interesante analizar estos casos para comprobar de qué microorganismo se trata. A la vista de estos resultados, parece que son necesarios otros enfoques para obtener

especies metagenómicas más robustas. Está planificado realizar una comparación de los dos resultados de *binning* mediante el programa DAS Tool (35), que por falta de tiempo no se ha podido incluir en este trabajo. En esta misma línea, se va a realizar también una comparación de los resultados del *binning* analizando la longitud y número de secuencias incluidas en los *bins* de Canopy y de MetaBAT2.

Este trabajo es el punto de partida para futuros estudios. El siguiente paso que se ha planificado consiste en reproducir un *pipeline* similar, pero en lugar de utilizar los genes, emplear los *contigs* directamente para realizar el *binning*. Otras posibilidades son reproducir el análisis desarrollado previamente por Pasolli *et al* (5), que utiliza la estrategia *single-sample* basada en *contigs*, o testear el resultado de co-ensamblaje múltiple para varias muestras.

Por otro lado, se plantea la posibilidad de emplear otros programas también basados en *binning* para analizar datos metagenómicos, como es ANVI'O (36) y Vizbin (37).

Otro de los pasos a realizar posteriormente consiste en mapear todas las secuencias con los genes de cada uno de los *bins* para, posteriormente realizar ensamblajes *single-bin* únicamente con los *reads* que hayan mapeado a cada *bin*, con el fin de obtener ensamblajes de mejor calidad. Sin embargo, con los resultados obtenidos, probablemente estos ensamblajes no sean tan buenos.

En cuanto a la forma de trabajar, se ha comprobado que para el manejo y análisis de datos metagenómicos un sistema computacional de alto rendimiento resulta fundamental. El disponer de los software necesarios comunes para todos los

usuarios del clúster, numerosos nodos y cores, así como la posibilidad de trabajar en remoto facilita el trabajo y permite optimizar el uso de los recursos.

5. Conclusiones

- Se ha obtenido un mejor ensamblaje de los datos metagenómicos con MetaSPAdes que con MEGAHIT tanto para muestras reales como para la *mock.*
- A pesar de que los resultados del *binning* no han sido buenos, con Canopy se han obtenido muchos más *bins* que con MetaBAT2, alguno de ellos de buena calidad.
- El trabajo en un clúster de PCs permite ahorrar tiempo de análisis y optimizar recursos.
- El procesamiento de los datos en paralelo en lugar de en serie, ahorra mucho tiempo de análisis.
- Son necesarios nuevos enfoques para conseguir mejores resultados, como la estrategia single-sample basada en *contigs*, con el objetivo de obtener especies metagenómicas fiables.

6. Valoración personal

Este trabajo me ha permitido no sólo aplicar todos los conocimientos adquiridos a lo largo del máster, sino además profundizar en algunos aspectos y aprender otros temas nuevos. Nunca antes había tenido la oportunidad de trabajar en un clúster ni con datos de *shotgun*. Estoy muy satisfecha con el trabajo realizado, ya que, aunque los resultados no han sido óptimos, he tenido la oportunidad de aprender a usar muchos programas diferentes, así como de entender el flujo de trabajo de este tipo de análisis que no es nada sencillo.

Todos los *scripts* están disponibles en GitHub y este trabajo es sólo el punto de partida para seguir trabajando en el diseño de un *pipeline* de análisis óptimo para este tipo de datos.

7. Glosario

Adaptador: secuencia de ADN que se une a los fragmentos de genoma que se van a secuenciar y que consta de tres partes: secuencia complementaria a las secuencias en la superficie donde se va a desarrollar la secuenciación; índice, diferente para cada muestra; y secuencia complementaria a los *primers* de secuenciación.

Alineamiento: proceso mediante el cual se analiza cómo y dónde unas secuencias determinadas son similares a una secuencia de referencia. Una "alineación" es el resultado de este proceso: una forma de "alinear" algunos o todos los caracteres de una secuencia con algunos caracteres de la referencia para comprobar cómo de similares son.

Bash: lenguaje de programación. Es un shell de Unix.

Binning: método por el que las secuencias obtenidas por *shotgun* se agrupan basándose en su composición y/o abundancia mediante diferentes algoritmos sin necesidad de una referencia.

Contig: segmentos de ADN superpuestos, que juntos representan una región consenso de ADN.

Clúster: conjunto de nodos que se comunican entre sí.

Co-ensamblaje: Ensamblaje de todas las secuencias de múltiples muestras, en lugar de hacer un ensamblaje independiente para cada muestra. Las ventajas que presenta son las siguientes: mayor cobertura; facilita la comparación entre muestras

al usar una referencia común para todos; y puede mejorar la recuperación de genomas a partir de datos metagenómicos gracias a las diferentes coberturas.

Conda: instalados de programas basados en Python y que contienen dependencias de Python.

Conexión VPN: Virtual Private Network o red privada virtual. Una conexión VPN te permite crear una red local sin necesidad de que sus integrantes estén físicamente conectados entre sí, sino a través de Internet.

Core: cada uno de los procesadores dentro de un nodo.

Ensamblaje: proceso mediante el cual secuencias cortas de ADN se vuelven a unir para crear una representación de los genomas originales a partir de los cuales se originó el ADN.

Especie metagenómica (metagenomic species, MGS): grupo de secuencias que se han agrupado por co-abundancia y que pueden corresponder a un taxón concreto.

Fedora: distribución Linux para propósitos generales basada en RPM como herramienta de administración de paquetes.

Librería: conjunto de secuencias de ADN de tamaños similares que llevan unidos los adaptadores y que es necesario para realizar la secuenciación.

L50: El menor número de contigs cuya longitud total representa al menos la mitad de la longitud del ensamblaje.

Metagenoma: conjunto de genes que contienen los microorganismos presentes en un determinado nicho ecológico.

Microbioma: conjunto de microorganismos en un determinado nicho ecológico, incluidos sus genes, metabolitos y ambiente.

Microbiota: conjunto de microorganismos presentes en un determinado nicho ecológico (ej. microbiota humana).

Mock: Una mezcla definida de bacterias y/o virus o bien de su ADN creadas *in vitro* para simular la composición de una muestra de microbiota o el ADN aislado de la misma.

N50: Longitud del contig más corto, en el conjunto de contigs más grandes que representa al menos el 50% del ensamblaje. Ejemplo:

PhiX: Bacteriófago que se utiliza como control en la secuenciación con Illumina y permite además introducir variabilidad en el caso de secuenciar librerías con poca diversidad, como por ejemplo en el caso de amplicones del gen 16S.

Pipeline: Conjunto determinado de pasos de procesamiento necesarios para convertir datos crudos en algo interpretable. Esto podría ser, por ejemplo, una serie de scripts o programas.

Read (secuencia): cada una de las secuencias que se obtienen del secuenciador.

Shell: Término usado en informática para referirse a un intérprete de comandos, el cual consiste en la interfaz de usuario tradicional de los sistemas operativos basados en Unix y similares, como GNU/Linux.

Secure shell o SSH: Protocolo de red que permite el intercambio de datos sobre un canal seguro entre dos computadoras. SSH usa técnicas de cifrado que hacen que la información que viaja por el medio de comunicación vaya de manera no legible y ninguna tercera persona pueda descubrir el usuario y contraseña de la conexión ni lo que se escribe durante toda la sesión. Se suele utilizar para iniciar una sesión en una máquina remota.

8. Referencias

1. D'Argenio V. Human Microbiome Acquisition and Bioinformatic Challenges in Metagenomic Studies. Int J Mol Sci 2018;19(2):383.

2. Breitwieser FP, Lu J, Salzberg SL. A review of methods and databases for metagenomic classification and assembly. Brief Bioinform 2017;1-15.

3. Quince C, Walker AW, Simpson JT, Loman NJ and Segata N. Shotgun metagenomics, from sampling to analysis. Nat Biotechnol 2017;35(9):833-44.

4. Thomas T, Gilbert J, Meyer F. Metagenomics - a guide from sampling to data analysis. Microb Inform Exp 2012;2(1):3.

5. Pasolli E, Asnicar F, Manara S, *et al*. Extensive Unexplored Human Microbiome Diversity Revealed by Over 150,000 Genomes from Metagenomes Spanning Age, Geography, and Lifestyle. Cell 2019;176(3):649-662.e20.

6. Mande SS, Mohammed MH, Ghosh TS. Classification of metagenomic sequences: methods and challenges. Brief Bioinform 2012;13(6):669-81.

7. Sedlar K, Kupkova K, Provaznik I. Bioinformatics strategies for taxonomy independent binning and visualization of sequences in shotgun metagenomics. Comput Struct Biotechnol J 2017;15:48–55.

8. Nielsen HB, Almeida M, Juncker AS, *et al*. Identification and assembly of genomes and genetic elements in complex metagenomic samples without using reference genomes. Nat Biotechnol 2014;32(8):822-8.

9. Wang Y, Leung HC, Yiu SM, *et al*. **MetaCluster** 5.0: a two-round binning approach for metagenomic data for low-abundance species in a noisy sample. Bioinformatics 2012;28:i356-62.

10. Wu YW, Simmons BA, Singer SW. **MaxBin** 2.0: an automated binning algorithm to recover genomes from multiple meta-genomic datasets. Bioinformatics 2016;32:605-7.

11. Imelfort M, Parks D, Woodcroft BJ, *et al*. **GroopM**: an automated tool for the recovery of population genomes from related metagenomes. PeerJ 2014;2:e603.

12. Alneberg J, Bjarnason BS, de Bruijn I, *et al*. Binning metagenomic contigs by coverage and composition. Nat Methods 2014;11:1144–6.

13. Kang DD, Froula J, Egan R, *et al*. **MetaBAT**, an efficient tool for accurately reconstructing single genomes from complex microbial communities. PeerJ 2015;3:e1165.

14. Guillén Y, Noguera-Julian M, Rivera J, *et al*. Low nadir CD4+ T-cell counts predict gut dysbiosis in HIV-1 infection. Mucosal Immunol 2019;12:232-46.

15. Andrews S. (2010) **FastQC**: a quality control tool for high throughput sequence data. Available online at: http://www.bioinformatics.babraham.ac.uk/projects/fastqc

16. Bolger AM, Lohse M, Usadel B. **Trimmomatic**: A flexible trimmer for Illumina Sequence Data. Bioinformatics 2014;30(15):2114-20.

17. Li D, Liu CM, Luo R, Sadakane K, and Lam TW. **MEGAHIT**: An ultra-fast single-node solution for large and complex metagenomics assembly via succinct de Bruijn graph. Bioinformatics 2015;31(10):1674-6.

18. Nurk S, Meleshko D, Korobeynikov A, and Pevzner PA. **MetaSPAdes**: a new versatile de novo metagenomics assembler. Genome Research 2017;27(5):824-34.

19. Mikheenko A, Saveliev V, Gurevich A. **MetaQUAST**: evaluation of metagenome assemblies. Bioinformatics 2016;32(7):1088-90.

20. Kang D, Li F, Kirton ES, *et al*. **MetaBAT2**: an adaptive binning algorithm for robust and efficient genome reconstruction from metagenome assemblies. PeerJ Preprints 2019;7:e27522v1.

21. Parks DH, Imelfort M, Skennerton CT, Hugenholtz P, Tyson GW. **CheckM**: assessing the quality of microbial genomes recovered from isolates, single cells, and metagenomes. Genome Research 2015;25:1043-55.

22. Ewels P, Magnusson M, Lundin S and Käller M. **MultiQC**: Summarize analysis results for multiple tools and samples in a single report. Bioinformatics 2016;32(19):3047-8.

23. Van der Walt A.J, van Goethem M.W, Ramond J.B, *et al*. **Assembling** metagenomes, one community at a time. BMC Genomics 2017;18:521.

24. Gurevich A, Saveliev V, Vyahhi N and Tesler G. **QUAST**: quality assessment tool for genome assemblies. Bioinformatics 2013;29(8):1072-5.

25. Quast C, Pruesse E, Yilmaz P, *et al*. The **SILVA** ribosomal RNA gene database project: improved data processing and web-based tools. Nucleic Acids Res 2013;41:590-6.Seemann T.

26. Camacho C, Coulouris G, Avagyan V, *et al*. **BLAST+**: architecture and applications. BMC bioinformatics 2009;10:421.

27. Seemann T. **Prokka**: Rapid Prokaryotic Genome Annotation. Bioinformatics 2014:15;30(14):2068-9.

28. Langmead B, Wilks C, Antonescu V, Charles R. Scaling read aligners to hundreds of threads on general-purpose processors. Bioinformatics 2018;35(3):421-432.

29. Langmead B, Salzberg SL. Fast gapped-read alignment with **Bowtie 2**. Nature Methods 2012;9(4):357-9.

30. Gordon A, Hannon GJ. (2010) **FastX Toolkit**. Available online at: http://hannonlab.cshl.edu/fastx_toolkit/index.html

31. Li H, Handsaker B, Wysoker A, *et al*. The Sequence alignment/map (SAM) format and **SAMtools**. Bioinformatics 2009;25:2078-9.

32. Huson DH, Beier S, Flade I, *et al*. **MEGAN** Community Edition - Interactive exploration and 2 analysis of large-scale microbiome sequencing data. PLoS Computat Biol 2016;12(6):e1004957.

33. Van der Walt AJ, Van Goethem MW, Ramond JB, *et al*. Assembling metagenomes, one community at a time. BMC Genomics 2017;18:521.

34. Vollmers J, Wiegand S, Kaster AK. Comparing and Evaluating Metagenome Assembly Tools from a Microbiologist's Perspective - Not Only Size Matters! PLOS ONE 2017; DOI:10.1371/journal.pone.0169662.

35. Sieber CMK, Probst AJ, Sharrar A, *et al*. Recovery of genomes from metagenomes via a dereplication, aggregation and scoring strategy. Nature Microbiol 2018;3:836-43.

36. Eren AM, Esen ÖC, Quince C, *et al*. Anvi'o: an advanced analysis and visualization platform for 'omics data. PeerJ 2015;3:e1319.

37. Laczny CC, Pinel N, Vlassis N, and Wilmen P. Alignment-free Visualization of Metagenomic Data by Nonlinear Dimension Reduction. Sci Rep 2014;4:4516.

9. Anexos

9.1 Scripts

https://github.com/avergarago/TFM_Vergara

9.2 MultiQC Report

https://drive.google.com/open?id=1jSp_rgm2PlgWLsZ_LFNmH4WmKFAwB_ms

9.3 Microorganismos incluidos en la mock

9.3.1 Información del producto

bei RESOURCES
SUPPORTING INFECTIOUS DISEASE RESEARCH

Product Information Sheet for HM-783D

Genomic DNA from Microbial Mock Community B (Staggered, Low Concentration), v5.2L, for 16S rRNA Gene Sequencing

Catalog No. HM-783D

For research use only. Not for human use.

Contributor:
Sarah K. Highlander, Associate Professor, Department of Molecular Virology and Microbiology; Baylor College of Medicine, Houston, Texas, USA

Product Description:
HM-783D contains genomic DNA from 20 bacterial strains containing staggered ribosomal RNA (rRNA) operon counts (1,000 to 1,000,000 copies per organism per L). This mock community is recommended for 16S rRNA gene sequencing by Sanger or amplicon sequencing methods. The recommended amount to use per experiment is 1 L.[1] The bacterial strains that DNA was extracted from are listed in Table 1.

Note: The label for HM-783D is incorrect. HM-783D contains genomic DNA from microbial mock community B and not microbial mock community A.

Table 1: Microbial Mock Community B

Organism	NCBI Reference Sequence
Acinetobacter baumannii, strain 5377	NC_009085
Actinomyces odontolyticus, strain 1A.21	NZ_AAYI02000000
Bacillus cereus, strain NRS 248	NC_003909
Bacteroides vulgatus, strain ATCC® 8482™	NC_009614
Clostridium beijerinckii, strain NCIMB 8052	NC_009617
Deinococcus radiodurans, strain R1 (smooth)	NC_001263, NC_001264
Enterococcus faecalis, strain OG1RF	NC_17316
Escherichia coli, strain K12, substrain MG1655	NC_000913
Helicobacter pylori, strain 26695	NC_000915
Lactobacillus gasseri, strain 63 AM	NC_008530
Listeria monocytogenes, strain EGDe	NC_003210
Neisseria meningitidis, strain MC58	NC_003112
Propionibacterium acnes, strain KPA171202	NC_006085
Pseudomonas aeruginosa, strain PAO1-LAC	NC_002516
Rhodobacter sphaeroides, strain ATH 2.4.1	NC_007493, NC_007494
Staphylococcus aureus, strain TCH1516	NC_010079
Staphylococcus epidermidis, FDA strain PCI 1200	NC_004461
Streptococcus agalactiae, strain 2603 V/R	NC_004116
Streptococcus mutans, strain UA159	NC_004350
Streptococcus pneumoniae, strain TIGR4	NC_003028

HM-783D has been qualified for PCR applications by amplification of approximately 1500 base pairs of the 16S rRNA gene.

Material Provided:
Each vial contains approximately 35 L of the bacterial genomic DNA mixture suspended in TE buffer (10 mM Tris-HCl and 1 mM EDTA, pH ~ 7.4). The concentration is shown on the Certificate of Analysis. The vial should be centrifuged prior to opening.

Packaging/Storage:
HM-783D was packaged aseptically in screw-capped plastic cryovials. The product is provided frozen on ice and should be stored at -20°C or colder immediately upon arrival. Freeze-thaw cycles should be minimized.

Citation:
Acknowledgment for publications should read "The following reagent was obtained through BEI Resources, NIAID, NIH as part of the Human Microbiome Project: Genomic DNA from Microbial Mock Community B (Staggered, Low Concentration), v5.2L, for 16S rRNA Gene Sequencing, HM-783D."

Biosafety Level: 1
Appropriate safety procedures should always be used with this material. Laboratory safety is discussed in the following publication: U.S. Department of Health and Human Services, Public Health Service, Centers for Disease Control and Prevention, and National Institutes of Health. Biosafety in Microbiological and Biomedical Laboratories. 5th ed. Washington, DC: U.S. Government Printing Office, 2009; see www.cdc.gov/biosafety/publications/bmbl5/index.htm.

Disclaimers:
You are authorized to use this product for research use only. It is not intended for human use.

BEI Resources
www.beiresources.org
E-mail: contact@beiresources.org
Tel: 800-359-7370
Fax: 703-365-2898

9.3.2 Lista de microorganismos incluidos como referencia

```
Acinetobacter_baumannii.fasta  Enterococcus_faecalis.fasta
Propionibacterium_acnes.fasta  Staphylococcus_epidermidis.fasta
Bacillus_cereus.fasta          Helicobacter_pylori.fasta
Pseudomonas_aeruginosa.fasta   Streptococcus_agalactiae.fasta
Bacteroides_vulgatus.fasta     Lactobacillus_gasseri.fasta
Rhodobacter_sphaeroides.fasta  Streptococcus_mutans.fasta
Clostridium_beijerinckii.fasta Listeria_monocytogenes.fasta
Schaalia_odontolytica.fasta*   Streptococcus_pneumoniae.fasta
Deinococcus_radiodurans.fasta  Neisseria_meningitidis.fasta
Staphylococcus_aureus.fasta
```

* Basónimo: *Actinomyces odontolyticus*

9.4 Resultados de CheckM

9.4.1 MetaBAT2/MEGAHIT

https://drive.google.com/open?id=1mQZ-E2Oylvs_e-NFTq6b-YEjuAG-MXiw

9.4.2 MetaBAT2/MetaSPAdes

https://drive.google.com/open?id=1PQdJibtCezHQHMaCEMJg-jB9ul3lwl4S

9.4.3 Canopy/MEGAHIT

https://drive.google.com/open?id=1QH5cWdUBpMGO2km7XL1ZB94Uh0TdzbCa

9.4.4 Canopy/MetaSPAdes

https://drive.google.com/open?id=1jErFozlTjBLeXierQoayxdIdu7QvM5sH

9.5 Programas

Se incluyen enlaces a los diferentes programas mencionados a lo largo del trabajo:

Anvi'o: http://merenlab.org/2016/06/22/anvio-tutorial-v2/#preparation

BLAST: https://www.ncbi.nlm.nih.gov/books/NBK279690/

Bowtie2: http://bowtie-bio.sourceforge.net/bowtie2/index.shtml

Canopy: https://bitbucket.org/HeyHo/mgs-canopy-algorithm/wiki/Home

CheckM: https://github.com/Ecogenomics/CheckM/wiki

DAST Tool: https://github.com/cmks/DAS_Tool

FastQC: https://www.bioinformatics.babraham.ac.uk/projects/fastqc/

FASTX-Toolkit: http://hannonlab.cshl.edu/fastx_toolkit/index.html

MEGAHIT: https://github.com/voutcn/megahit

MEGAN: http://ab.inf.uni-tuebingen.de/data/software/megan6/download/welcome.html;

http://ab.inf.uni-tuebingen.de/data/software/megan6/download/manual.pdf

MetaBAT: https://bitbucket.org/berkeleylab/metabat/wiki/browse/

MetaBAT2: https://bitbucket.org/berkeleylab/metabat/src/master/

MetaSPAdes: http://cab.spbu.ru/software/meta-spades/

MultiQC: https://multiqc.info/

PROKKA: https://github.com/tseemann/prokka

QUAST: http://quast.sourceforge.net/quast

SAMTOOLS: http://samtools.sourceforge.net/

SRA Tools: https://ncbi.github.io/sra-tools/

Trimmomatic: http://www.usadellab.org/cms/?page=trimmomatic

VizBin: http://claczny.github.io/VizBin/