Estudio comparativo de herramientas bioinformáticas para el análisis metagenómico sobre el microbioma humano

Lidia Jiménez Ocón
Máster en Bioinformática y Bioestadística
Área 3.9: Análisis e integración de datos ómicos

Andreu Paytuví Gallart
Ferran Prados Carrasco

08/01/2020

FICHA DEL TRABAJO FINAL

Título del trabajo:	*Estudio comparativo de herramientas bioinformáticas para el análisis metagenómico sobre el microbioma humano.*
Nombre del autor:	*Lidia Jiménez Ocón*
Nombre del consultor/a:	*Andreu Paytuví Gallart*
Nombre del PRA:	*Ferran Prados Carrasco*
Fecha de entrega (mm/aaaa):	01/2020
Titulación:	Máster en Bioinformática y Bioestadística
Área del Trabajo Final:	Análisis e integración de datos ómicos
Idioma del trabajo:	*Castellano*
Palabras clave	Metagenómica, microbioma, 16S rRNA.

Resumen del Trabajo:

La finalidad de este trabajo final de Máster ha sido realizar una comparativa con diferentes softwares para el análisis taxonómico en diferentes microbiomas del cuerpo humano.

El estudio metagenómico se llevó a cabo gracias al método secuenciación por amplicón del gen 16S rRNA, este es marcador por excelencia caracterizado por su baja tasa de evolución.

Para hacer el estudio se crearon tres dataset correspondientes a microbioma intestinal, vaginal y bucal. La simulación fue realizada con el software Wgsim con datos del MiSeq de Illumina, llevando a cabo una secuenciación paired-end.

El análisis bioinformático se realizó con tres de los softwares más utilizados en la actualidad, que son: Gaia, Qiime2 y mothur. Estos dos últimos son de código abierto por lo que para llevarlos a cabo se hizo uso de una máquina virtual de Ubuntu en Google cloud. Gaia ha sido desarrollado por la empresa Sequentia Biotech, por lo que no hubo necesidad de crear ningún código para el análisis.

Una vez realizado el análisis taxonómico se llevó a cabo el análisis estadístico de los resultados. Para ello se evaluaron parámetros como precisión, recall y F-measure. Se comprobó que el software que ofrecía mejores resultados para cada uno de los parámetros evaluados en el nivel taxonómico de género era Gaia.

Abstract:

The purpose of this Master's final project has been to perform a benchmarking using software for a taxonomic analysis in different microbiomes of the human body.

This metagenomic study was carried out thanks to the amplicon sequencing methods of the 16S rRNA gene, this is a marked par excellence characterized by its low rate of evolution.

In order to make this study three datasets corresponding to intestinal, vaginal and oral microbiomes were created. The simulation was conducted with Wgsim software using data from the Miseq benchtop sequencer by Illumina, applying a Paired-End Sequencing.

The bioinformatic analysis was carried out with three of the most commonly used softwares nowadays, which are Gaia, Qiime2 and Mothur. These last two are open sourced so in order to execute them we used an Ubuntu Virtual Machine in Google Cloud. Gaia has been developed by the company Sequentia Biotech so there was no need to create any code for the analysis.

Once the taxonomic analysis was performed the statistical analysis of the results was implemented. For this, parameters such as precision, recall and F-measure were evaluated. It was found that Gaia was the software that offered the best results for each of the parameters evaluated at the taxonomic level of gender.

Índice

Lista de figuras

Lista de tablas

1. Introducción

1.1 Contexto y justificación del trabajo

El presente Trabajo Fin de Máster consistirá en realizar una comparativa de análisis Metagenómico con diferentes herramientas bioinformáticas. Ello se hará con datos del microbioma humano. Denominamos microbioma al conjunto de genes de los microorganismos presentes en el cuerpo humano, asociándose e interactuando con este. Debido a que son comunidades microbianas, no se puede separar un microorganismo de otro, por lo que se estudiarán en su conjunto. La metagenómica nos permite estudiar poblaciones bacterianas y así poder entender cuál es el papel que desempeñan en el medio donde se encuentran [1].

El estudio metagenómico se puede llevar a cabo gracias a dos métodos: secuenciación del genoma completo (WGS) o "shotgun" y secuenciación por amplicones del gen 16S rRNA. En el caso de la secuenciación del genoma completo, su análisis conlleva un mayor coste debido a que en lugar de adaptarse a un locus genómico específico para la amplificación, todo el ADN se cortaría en pequeños fragmentos que serían secuenciados de forma independiente. Contribuiría a la obtención de información funcional y genética, identificando los diferentes microorganismos presentes en la muestra. Este necesita mayor tiempo de procesado de análisis. La secuenciación por amplicones, es bastante más sensible debido a que el gen 16S es exclusivo de cada especie, reduce los costes y el tiempo de respuesta. Una vez que se han obtenido las secuencias de la región del metagenoma 16S, el siguiente paso sería estimar la diversidad microbiana y la composición taxonómica [17].

La elección de este tema radica en la importancia de su estudio. El número de estudios del microbioma aumenta exponencialmente ya que se ha visto relación entre las comunidades microbianas y algunas enfermedades. Por ejemplo, hay estudios en donde se analizan los cambios del microbioma en distintas condiciones con respecto a la dieta, el uso de antibióticos, enfermedades, edad, localización geográfica, etc. Incluso factores externos como el estilo de vida y la frecuencia de ejercicio también se ven afectados. Dado que el microbioma tiene un gran impacto en la salud humana, conocer el microbioma nos podría ayudar a prevenir enfermedades. Aparte de las aplicaciones clínicas, también hay aplicaciones industriales y medioambientales [2].

1.2 Objetivos del trabajo

Como objetivos generales a cubrir tenemos los siguientes:

1.2.1. Objetivos generales:

- **Objetivo 1**: Obtener los datasets artificiales.
- **Objetivo 2**: Utilización de herramientas bioinformáticas.
- **Objetivo 3**: Control de calidad de los resultados.

Estos objetivos darán lugar a estos otros que son específicos:

1.2.2. Objetivos específicos:

- **Objetivo 1**: Obtener los datasets artificiales.
  a) Obtener genomas de la microbiota humana.
  Se seleccionarán distintas especies de la microbiota intestinal, bucal y vaginal. Se obtendrá su secuencia genómica en formato fasta.
  b) Generar los reads de cada genoma.
  Una vez que tenemos la secuencia de cada una de las especies es necesario generar los reads. Para ello, vamos a hacer uso del simulador "Wgsim". Lo que vamos a simular son los resultados de secuenciación del equipo MiSeq de la plataforma Illumina.
- **Objetivo 2:** Utilización de herramientas bioinformáticas.
  a) Ejecución de los diferentes softwares para el análisis.

Se van a utilizar tres softwares: Gaia, Qiime2 y Mothur.

- **Objetivo 3:** Control de calidad de los resultados.
  a) Análisis estadístico de los datos obtenidos. Para ello se calculará la sensibilidad, precisión y F-score.
  b) Establecer unas conclusiones a partir de los resultados.
  Una vez evaluados los resultados, analizaremos que software nos parece más fiable a la hora de tratar los datos.

1.3 Enfoque y método seguido

Para la realización de las tareas, inicialmente se creó una máquina virtual en Windows. Debido a su bajo rendimiento se decidió hacer una partición en el disco duro y se instaló el sistema operativo Ubuntu (versión 19.04), siendo el utilizado finalmente para la realización de todas las tareas. Debido a los problemas de bajo rendimiento computacional se tuvo que configurar una instancia de una máquina virtual de Google Cloud. Dicha maquina está compuesta por 15 GB de RAM y 250GB de disco duro HDD. Esta fue la estrategia más apropiada a seguir debido a que teníamos un gran volumen de datos para procesar, y el uso de un ordenador doméstico se quedaba corto de memoria.

1.4 Planificación del trabajo

La siguiente tabla muestra la planificación inicial de las tareas llevadas a cabo durante el periodo de realización del TFM.

Ítem	Actividad	Tarea	Inicio	Fin
1	Definición de los contenidos del trabajo	PEC0	2019.09.18	2019.09.30
2	Desarrollo del Plan de Trabajo	PEC1	2019.10.01	2019.10.14
3	Obtención de genomas	PEC2	2019.10.15	2019.10.20
4	Simulación de reads	PEC2	2019.10.21	2019.10.28
5	Utilización de las herramientas metagenómicas	PEC2	2019.10.29	2019.11.18
6	Análisis metagenómico de los datos	PEC3	2019.11.19	2019.12.02
7	Análisis de resultados	PEC3	2019.12.03	2019.12.16
8	Cierre de la memoria	PEC4	2019.12.17	2020.01.08
9	Elaboración de la presentación	PEC5	2020.01.09	2020.01.12
10	Defensa TFM	PEC6	2020.01.15	2020.01.22

Tabla 1: Planificación inicial de las tareas

Además, se muestra el diagrama de Gantt. Este ha sido realizado con el programa R [4].

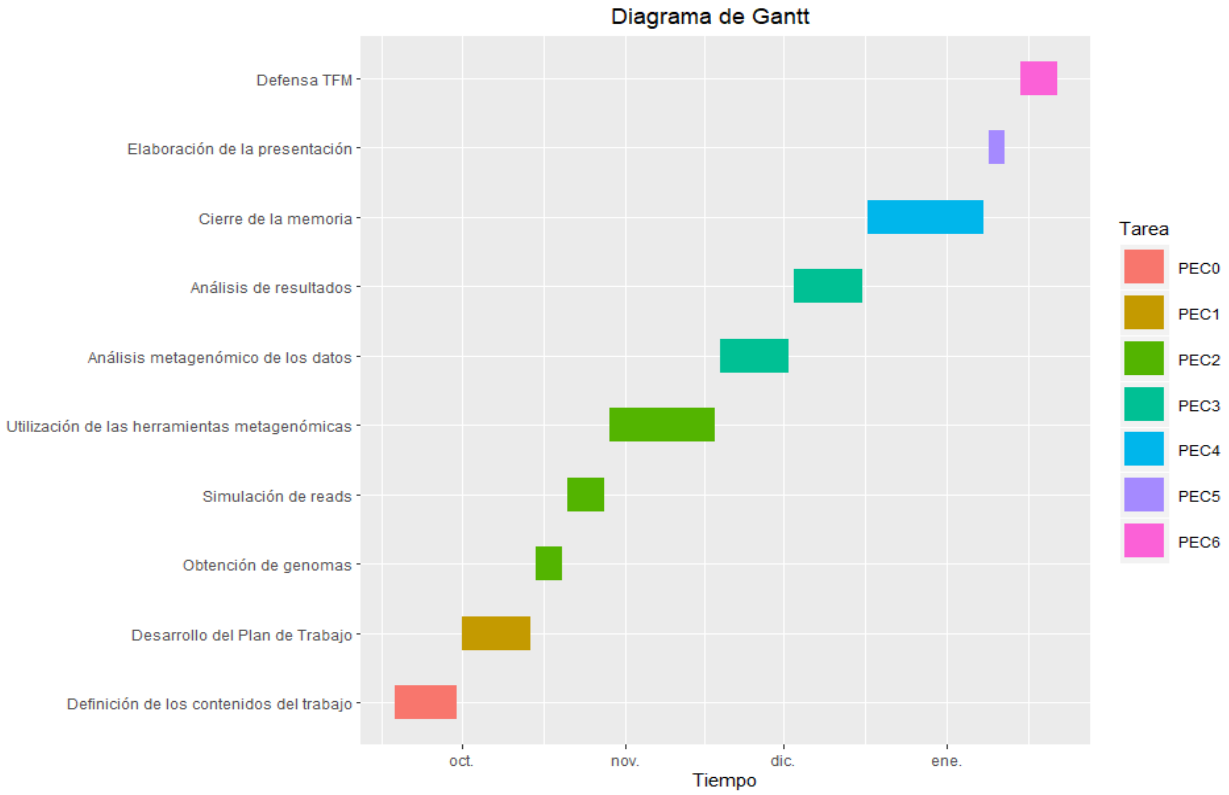

Figura 1: Diagrama de Gantt de planificación inicial

Finalmente, la planificación inicial fue modificada conforme se iba avanzando con las tareas propuestas. Se realizó un diagrama de Gantt final con las fechas asociadas a cada una de ellas.

Ítem	Actividad	Tarea	Inicio	Fin
1	Definición de los contenidos del trabajo	PEC0	2019.09.18	2019.09.30
2	Desarrollo del Plan de Trabajo	PEC1	2019.10.01	2019.10.14
3	Obtención de genomas	PEC2	2019.10.15	2019.10.20
4	Instalación Wgsim. Simulación de reads	PEC2	2019.10.21	2019.10.30
5	Instalación QIIME2	PEC2	2019.10.31	2019.11.04
6	Práctica con QIIME2	PEC2	2019.11.05	2019.11.15
7	Preparación PEC2	PEC2	2019.11.16	2019.11.18
8	Obtención genoma Vaginal y Bucal	PEC3	2019.11.19	2019.11.24
9	Análisis con QIIME2 y GAIA	PEC3	2019.11.25	2019.12.05
10	Análisis con mothur	PEC3	2019.12.06	2019.12.16
11	Cierre de la memoria	PEC4	2019.12.17	2020.01.08
12	Elaboración de la presentación	PEC5	2020.01.09	2020.01.12
13	Defensa TFM	PEC6	2020.01.15	2020.01.22

Tabla 2: Planificación final de las tareas

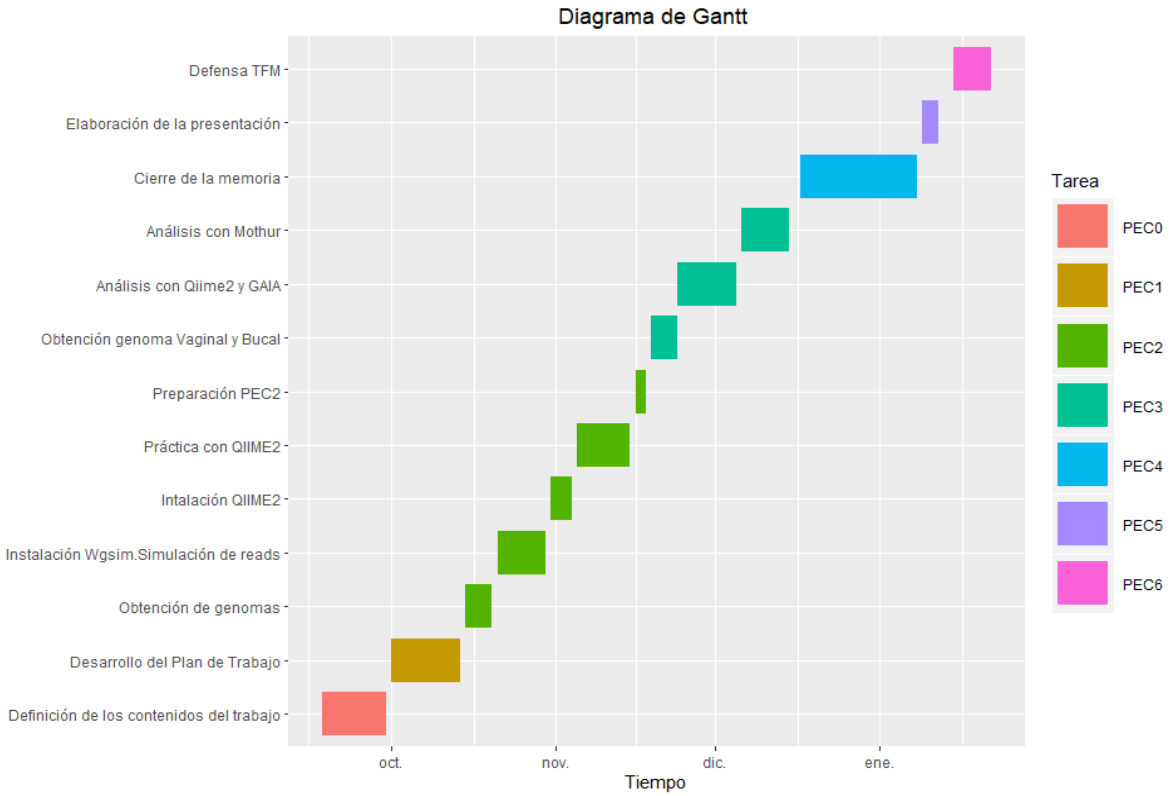

Figura 2: Diagrama de Gantt final

1.5 Breve sumario de productos obtenidos

Una vez finalizado el TFM se han generado los siguientes documentos:

- Plan de trabajo

El plan de trabajo es un documento en donde se refleja el contexto y justificación del trabajo junto con los objetivos a conseguir y la planificación cronológica llevada a cabo.

- Memoria

La memoria es el documento en el cual se refleja todo el trabajo realizado. El plan de trabajo formará parte de la memoria junto con los resultados y las conclusiones obtenidas.

- Presentación virtual

Se hará un resumen del trabajo realizado y se presentaran las diapositivas mediante Microsoft Power Point.

- Autoevaluación del proyecto

Esta consistirá en una evaluación del resultado final

1.6 Breve descripción de los otros capítulos de la memoria

Se explica muy brevemente el contenido los capítulos que componen la memoria:

- Introducción

Su finalidad es poner en contexto el tema elegido con la temática a resolver.

- Metodología

Este capítulo hace un recorrido desde la obtención de los datos crudos, su simulación y tratamiento, hasta el análisis con los diferentes softwares.

- Resultados

Una vez realizado el análisis con los tres softwares, en este apartado se tratarán los resultados, y se valorará cuál de ellos nos ofrece una mayor fiabilidad.

- Conclusión

Finalmente, en este apartado se llegará a una valoración que nos permitirá tomar decisiones respecto al tema tratado durante el Trabajo.

2. Metodología

Como ya se ha especificado anteriormente, lo que se va a tratar serán datos generados in silico. Vamos a simular datos de la plataforma Illumina, en concreto del equipo Miseq System [11]. Illumina es una de las compañías líderes en tecnologías NGS o secuenciación de alto rendimiento. Desarrolla sistemas que ofrece grandes ventajas en comparación con las técnicas tradicionales, ya que aumenta el rendimiento de los procesos, son más sensibles y disminuyen los costes de producción. Esta tecnología proporciona los servicios necesarios a los mercados de secuenciación, genotipado y expresión génica. Además, de la capacidad para realizar análisis genéticos, a partir de los cuales podremos extraer información para el análisis de variación genética y función biológica [10].

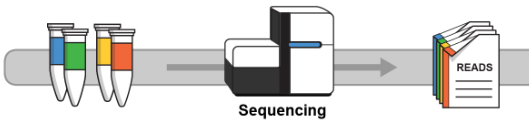

Figura 3: Imagen del proceso de secuenciación [20]

MiSeq System es uno de los equipos que ofrece Illumina, es una plataforma de secuenciación de ADN, la cuál es capaz de convertir el resultado en un conjunto de datos. Ofrece amplificación, secuenciación y análisis de datos en un único instrumento. Es un equipo de pequeñas dimensiones, por lo que se puede acoplar en cualquier entorno de trabajo de un laboratorio. MiSeq System utiliza la tecnología de secuenciación por síntesis (SBS) de Illumina, los procesos químicos de secuenciación más usados en todo el mundo. Con la potencia de la secuenciación de segunda generación y un tamaño reducido, MiSeq es la plataforma idónea para análisis genético rápido y rentable [11]. Aparte del Miseq de Illumina, también hay otras compañias que desarrollan este tipo de secuenciación como el equipo 454 GS Junior de Roche, o el Ion Torrent de Life Technologies.

La secuenciación por síntesis se lleva a cabo en 4 etapas [11]:

- Preparación de la genoteca

La genoteca se prepara fragmentando una muestra de cDNA o gDNA y ligando adaptadores específicos (5’ y 3’) a ambos extremos del fragmento.

- Generación del clúster

La genoteca se carga en una celda de flujo (flow cell) y los fragmentos se van hibridando con la superficie de la celda. Cada fragmento unido a la superficie de la celda se amplifica a través de una amplificación por puente. Una vez que el clúster está completo, la plantilla está lista para la secuenciación.

- Secuenciación por síntesis

Se añaden los reactivos de secuenciación, incluidos nucleótidos marcados con fluorescencia, y se incorpora la primera base. Se toma una imagen de la celda de flujo y se registra la emisión de cada grupo. La longitud de onda de emisión y la intensidad se utilizan para identificar la base. Este ciclo se repite n veces para crear una longitud de lectura de n bases.

- Análisis de datos

Las lecturas se alinean a una secuencia de referencia con un software bioinformático. Después de la alineación, se puede hacer una comparativa entre el genoma de referencia y las lecturas (reads) obtenidas.
En la figura 4 podemos ver los tres primeros pasos de la secuenciación por síntesis.

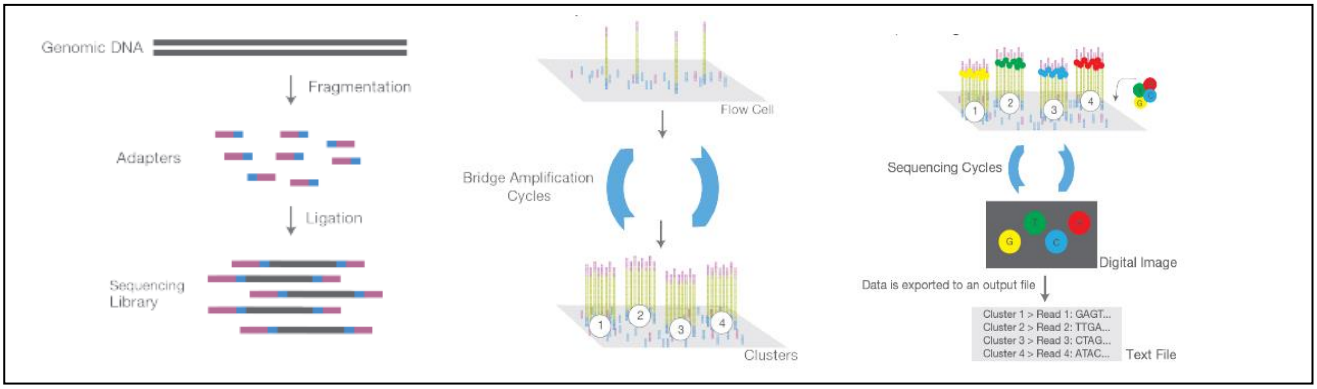

Figura 4: Secuenciación por síntesis [11]

2.1 Obtención de los Dataset

Los estudios metagenómicos se realizan normalmente mediante el análisis del gen de ARN ribosómico 16S procariótico (16S rRNA). Este contiene nueve regiones variables (V1-V9) intercaladas entre regiones conservadas. Las regiones variables del gen 16S rRNA se utilizan con frecuencia en clasificaciones filogenéticas debido a sus bajas tasas de evolución. La región de interés dependerá del objetivo experimental y del tipo de muestra. El gen 16S es un componente de la subunidad 30S de los ribosomas procariotas. Es la macromolécula más ampliamente utilizada como marcador en estudios de filogenia y taxonomía bacterianas [2],[3].

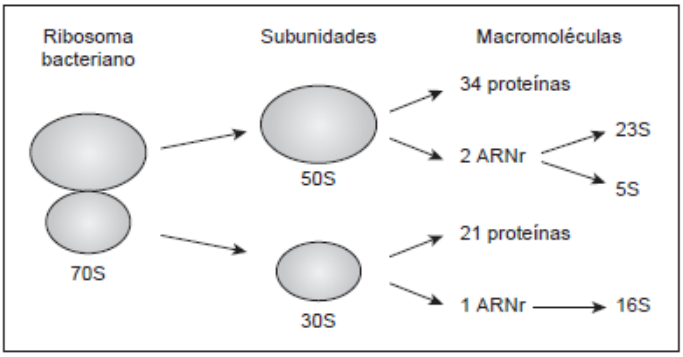

Figura 5: Esquema del ribosoma bacteriano. [2]

Para realizar el análisis que se detalla en el presente trabajo se obtuvieron las secuencias de la región 16S rRNA.

2.1.1 Microbiota intestinal:

La microbiota intestinal se compone de un conjunto de especies de microorganismos, incluidas bacterias, levaduras y virus. Entre la división taxonómica bacteriana que tenemos en la microbiota intestinal, dominan los *Firmicutes, Bacteroidetes, Actinobacteria, Proteobacteria, Fusobacteria y Verrucomicrobia*, con la divisiones de *Firmicutes* y *Bacteroidetes* representando el 90% de microbiota intestinal [1],[7].

La microbiota intestinal de cada individuo se caracteriza específicamente por grupos de bacterias llamadas enterotipos. La composición bacteriana se clasifica en tres grupos:

- Enterotipo I: predominan los *Bacteroides*, está asociado a una dieta rica en grasas y proteínas animales.
- Enterotipo II: predominancia de *Prevotella*, asociado a una dieta rica en carbohidratos.

- Enterotipo III: en la que predominan los *Ruminococcus*, es el enterotipo predominante en la población.

Los enterotipos parecen estar definidos principalmente por los hábitos alimenticios. Comprender los orígenes y las funciones de los enterotipos puede mejorar el conocimiento de las relaciones entre la microbiota intestinal y la salud humana.[1]
Esta imagen representa de una manera muy visual la composición de la microbiota.

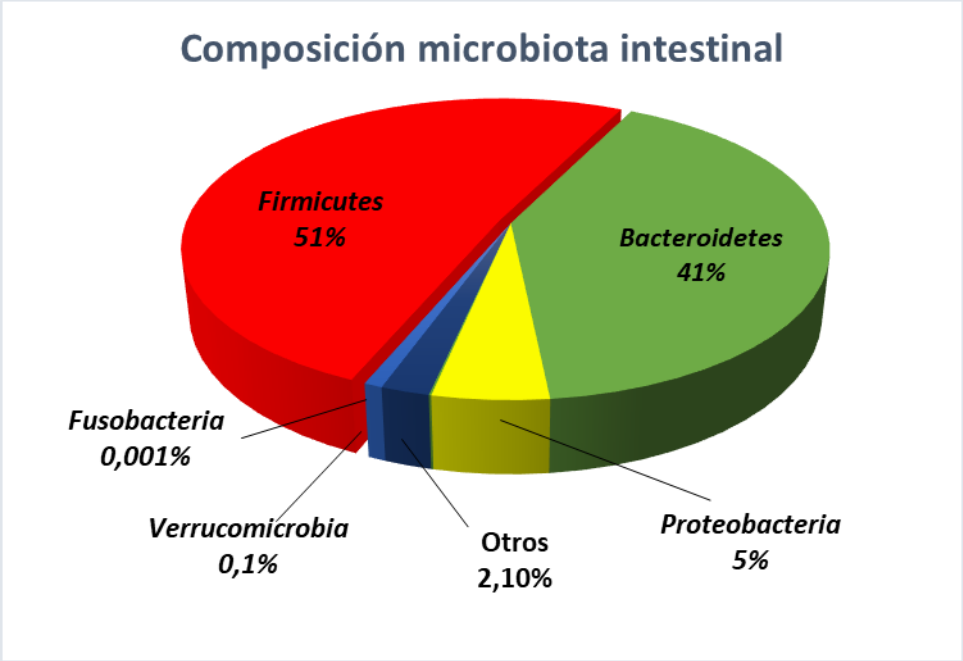

Figura 6: composición microbiota intestinal

Para la obtención del dataset se ha realizado una búsqueda bibliográfica. Se han elegido distintas especies representativas, ajustándose a una proporción estimada en el cuerpo humano de una persona adulta y sana. Es una simulación de datos, por lo que tanto las especies como la proporción son variables en la realidad. Se hace una estimación de alrededor de unos cien mil reads por dataset, por lo que se divide ajustándose a su proporción entre las especies seleccionadas. En esta tabla se puede apreciar las especies elegidas:

Phylum	Proporción phylum	Especie	Proporción	Reads	Total reads
Actinobacteria	0,80 %	*Bifidobacterium longum*	0,40 %	400	800
		Bifidobacterium bifidum	0,40 %	400	
Firmicutes	51 %	*Faecalibacterium prausnitzii*	0,5%	255	51000
		Anaerostipes hadrus	23,50 %	12062	
		Dorea longicatena	24,00 %	12063	
		Fusicatenibacter saccharivorans	23,50 %	12062	
		Blautia obeum	24 %	12063	
		Roseburia intestinalis	0,5%	255	
		Ruminococcus faecis	1%	510	
		Dialister invisus	1%	510	
		Lactobacillus reuteri	1%	510	
		Enterococcus faecium	1%	510	

Bacteroidetes	41 %	*Bacteroides fragilis*	5 %	5000	41000
		Bacteroides vulgatus	5 %	5000	
		Bacteroides uniformis	5 %	5000	
		Parabacteroides distasonis	5 %	5000	
		Alistipes finegoldii	5 %	5000	
		Prevotella colorans	8 %	8000	
		Prevotella buccae	8%	8000	
Proteobacteria	5 %	*Escherichia coli*	2 %	2000	5000
		Shigella flexneri	1 %	1000	
		Desulfovibrio intestinalis	1 %	1000	
		Bilophila wadsworthia	1 %	1000	
Fusobacteria	0,001 %	*Fusobacterium nucleatum*	0,001 %	1	1
Verrucomicrobia	0,10 %	*Akkermansia muciniphila*	0,10 %	100	100
Otros	2,10 %				2100
Total reads				100000	

Tabla 3: Especies en la microbiota intestinal

2.1.2 Microbiota vaginal:

El microbioma vaginal está compuesto por más de 200 especies bacterianas en la que predomina el filo *Firmicutes* y *Lactobacillus* como género. La función principal de la microbiota vaginal es defender al organismo de las infecciones. Su composición depende en gran medida de origen étnico, genético, edad, embarazo, etc. Un desequilibrio en el microbioma vaginal, conocido como disbiosis, puede llegar a dar lugar a infecciones vaginales entre otros problemas [14].

Para obtener los genomas de los microorganismos de la microbiota vaginal humana, se hizo una búsqueda bibliográfica y se seleccionaron algunas de las especies más representativas obteniéndose su proporción de manera aproximada. Al igual que en el caso anterior, son datos simulados por lo que su proporción puede distar de la realidad. En la siguiente figura podemos apreciar su composición.

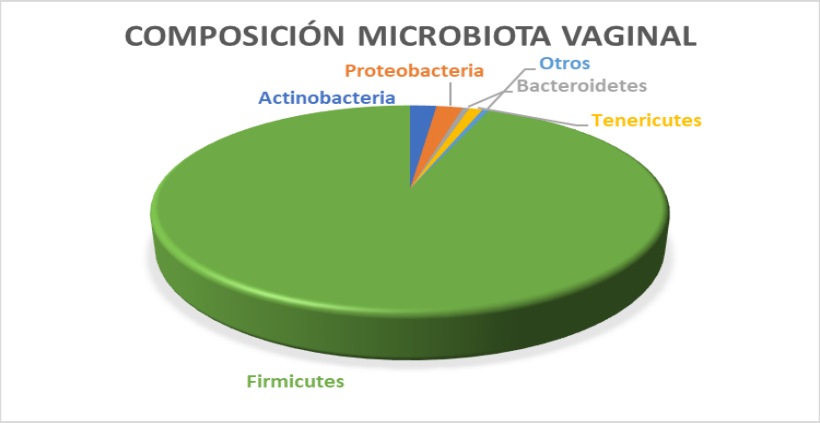

Figura 7: Composición microbiota vaginal

En la siguiente tabla se observan las especies elegidas para el análisis:

Phylum	Proporción phylum	Especie	Proporción	Reads	Total
Firmicutes	94%	*Lactobacillus crispatus*	14%	14000	94000
		Lactobacillus iners	10%	10000	

		Lactobacillus jensenni	14%	14000	
		Lactobacillus gasseri	14%	14000	
		Lactobacillus salivarius	5%	5000	
		Lactobacillus vaginalis	5%	5000	
		Lactobacillus rhamonsus	4%	4000	
		Lactobacillus casei	5%	5000	
		Lactobacillus plantarum	5%	5000	
		Lactobacillusacidophilus	4%	4000	
		Lactobacillus fermentum	5%	5000	
		Lactobacillus brevis	5%	5000	
		Staphylococcus epidermidis	4%	4000	
Actinobacteria	2%	*Bifidobacterium longum*	1%	1000	2000
		Bifidobacterium bifidum	1%	1000	
Proteobacteria	2%	*Neisseria lactamica*	1%	1000	2000
		Escherichia coli	1%	1000	
Bacteroidetes	0,50%	*Prebotella bivia*	0,5	500	500
Tenericutes	1%	*Mycoplasma hominis*	1%	1000	1000
Otros	0,50%	No se conoce			
					99500

Tabla 4: Especies en la microbiota vaginal

2.1.3 Microbiota bucal:

Las técnicas de secuenciación han permitido conocer la composición de las comunidades de microorganismos presentes en la microbiota bucal. Se estima que el número de filotipos está alrededor de 19000. Si tenemos en cuenta que la cavidad oral es la principal puerta de entrada a posibles infecciones, es de vital importancia comprender como está constituida. Esto es una tarea compleja ya que existen una gran variedad de hábitats en la mucosa oral. Estas serán dependientes de la disponibilidad de nutrientes, de la concentración de oxígeno, junto con la exposición a factores inmunológicos y las características anatómicas de la persona en cuestión.

La metagenómica es la mejor herramienta disponible capaz de definir su composición. Ello es llevado a cabo a través de la amplificación del gen 16S rRNA como ya se ha comentado anteriormente. Dichos estudios poblacionales demuestran que los filos de la cavidad bucal son: *Firmicutes, Proteobacteria, Bacteriodes, Actinobacteria y Fusobacteria*. Siendo los *Firmicutes* el filo predominante en esta cavidad [15].

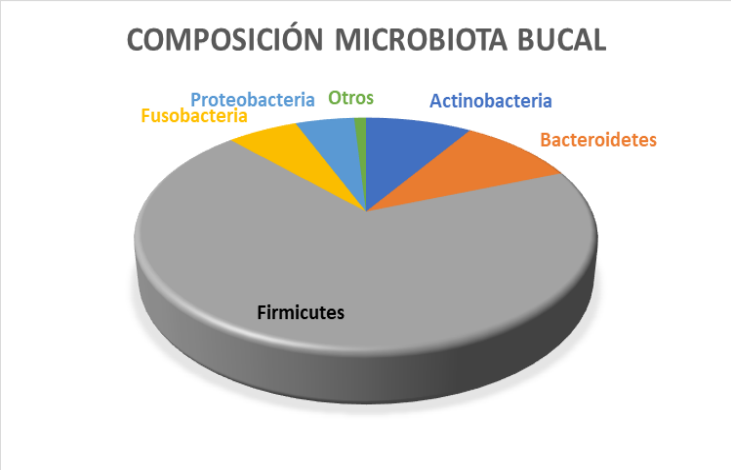

Figura 8: Composición microbiota bucal

En la siguiente tabla, al igual que en los datasets anteriores, se ha realizado una búsqueda bibliográfica para identificar las especies predominantes en la microbiota bucal.

Phylum	Proporción phylum	Especie	Proporción	Reads	Total
Actinobacteria	9,00%	Corynebacterium durum	3%	3000	9000
		Rothia dentocariosa	3%	3000	
		Rothia mucilaginosa	3%	3000	
Bacteroidetes	10%	Porphyromonas endodontalis	2%	2000	10000
		Prevotella pallens	2%	2000	
		Prevotella nigrescens	2%	2000	
		Prevotella nanceiensis	2%	2000	
		Prevotella melaninogenica	2%	2000	
Firmicutes	69%	*Streptococcus mutans*	12%	12000	69000
		Streptococcus intermedius	12%	12000	
		Streptococcus oralis	12%	12000	
		Streptococcus sanguinis	12%	12000	
		Streptococcus agalactiae	5%	5000	
		Veillonella parvula	4%	4000	
		Veillonella atypica	4%	4000	
		Veillonella denticariosi	4%	4000	
		Veillonella dispar	4%	4000	
Fusobacteria	6%	*Fusobacterium nucleatum*	6%	6000	6000
Proteobacteria	5,00%	*Haemophilus parainfluenzae.*	2,50%	2500	5000
		Neisseria flavescens	2,50%	2500	
Otros	1,00%	No se conoce			
					Total 99000

Tabla 5: Especies en la microbiota bucal

2.2 Obtención de secuencias

Una vez que tenemos las especies seleccionadas, el siguiente paso sería obtener la secuencia de cada una de ellas en formato fasta. Este formato se caracteriza por tener una primera línea que describe la secuencia, identificándose por el símbolo " > ", seguida por líneas de secuencia de nucleótidos. Su formato está basado en un texto plano [9].
Para ello, accedimos a la página del NCBI (National Center for Biotechnology) (https://www.ncbi.nlm.nih.gov/). Filtramos por "nucleotide" añadiendo la especie a buscar y obtenemos la secuencia en formato fasta de cada una de las especies.

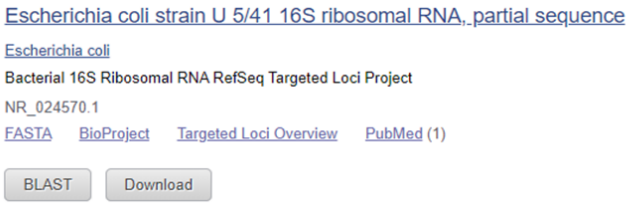

Figura 9: Obtención de secuencias fasta

En esta imagen observamos las características del formato fasta:

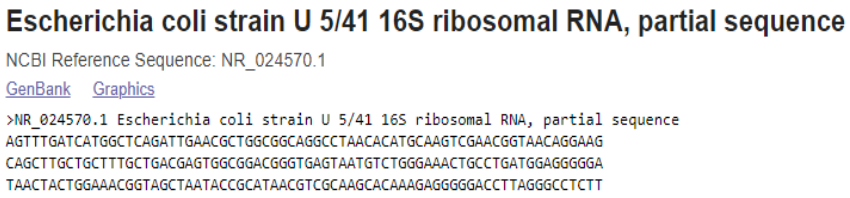

Figura 10: Imagen formato fasta

2.3 Simulación de reads

Una vez que tenemos las secuencias seleccionadas en formato fasta, vamos a hacer la simulación de los reads. De manera que para el desarrollo de este trabajo simularíamos los resultados de la plataforma Illumina, en concreto el de su secuenciador Miseq. Ello lo llevaremos a cabo utilizando el programa Wgsim.

De este programa se obtendrá por cada secuencia analizada dos archivos en formato fastq. Este formato tiene unas determinadas características, que vienen descritas en cuatro líneas [6], que son:

- Línea 1: empieza por la letra arroba (@). Es el identificador de la secuencia, y en esa línea se describe información importante con datos de la secuencia.
- Línea 2: corresponde a la secuencia sin procesar.
- Línea 3: contiene el carácter "+". Se le llama identificador.
- Línea 4: codifica los valores de calidad para la secuencia en la línea 2 y debe contener el mismo número de símbolos que letras en la secuencia. Los caracteres mostrados representan un valor numérico que se traduce como un puntaje de calidad Phred.

Los puntajes de calidad Phred se utilizan para evaluar la calidad de la secuencia. El valor numérico asociado a cada una de las bases es un valor de probabilidad de que la base se haya secuenciado erróneamente. Se detalla en la siguiente fórmula:

$$Q = -10\log_{10}P$$

Siendo Q el valor numérico y P la probabilidad de error de la secuenciación. Por ejemplo, si Phed asigna un puntaje de calidad de 20 a una base, la posibilidad de que esa base sea nombrada incorrectamente es de 1 en 100. Por lo que nos interesa que el valor de Q sea elevado, eso se traduce en un bajo valor de P y por tanto la secuenciación de esa base ha sido de buena calidad. [12]
En cada una de nuestras secuencias se ha establecido puntajes de calidad base de I, que es, equivalente a un Sanger Phred + 33 puntaje de 40.

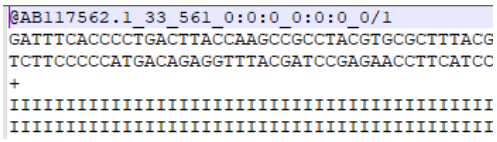

Figura 11: Imagen formato fastq

La simulación de los datos se llevó a cabo gracias al desarrollo de la Secuenciación “Paired-End”. La secuenciación Paired-End implica secuenciar ambos extremos de los fragmentos de ADN y alinear los read forward y reverse como lectura emparejada. Además de producir el doble de lecturas por el mismo tiempo y esfuerzo en la preparación de la librería, las secuencias alineadas permiten una alineación más precisa. Este análisis también permite la eliminación de duplicados de PCR, un artefacto común resultante de amplificación durante la preparación de la librería [11].

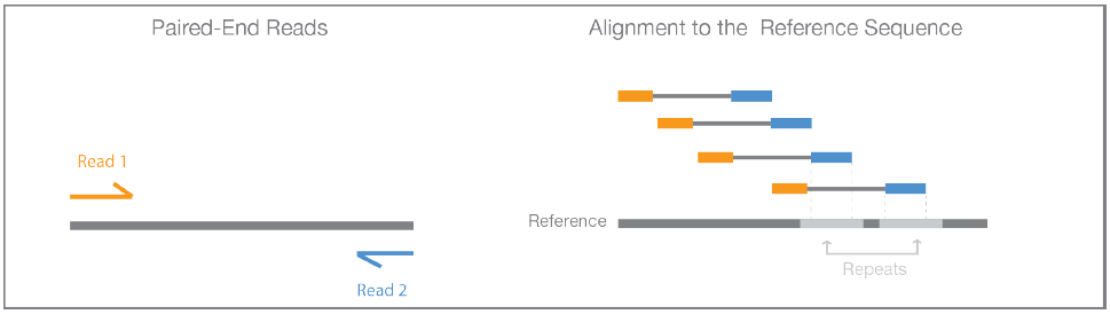

Figura 12: Paired-End Reads [11]

Para hacer la simulación de reads utilizamos el software Wgsim. Esta es una herramienta que se ha obtenido a través del repositorio de archivos GitHub. Se ejecuta a través de la línea de comandos de Ubuntu. Siguiendo las instrucciones del programa Wgsim, vamos modificando los parámetros para obtener una buena simulación [5].
Para comenzar a trabajar con Wgsim tenemos que tener claro el número de reads que vamos a generar, en este caso realizaremos sobre unas 100000 reads por dataset. En cada una de las tablas anteriores se ha especificado el número de reads por especie.
Ahora que ya tenemos de manera aproximada la proporción del número de reads, es el momento de hacer la simulación, para ello hacemos uso de la siguiente instrucción:

wgsim -1pb -2pb -d -r -e -N -R -X especie.fa especie.read1.fq especie.read2.fq

Cada uno de los parámetros corresponde a lo siguiente:
- pb: Para Illumina normalmente se utiliza la máquina MiSeq obteniendo paired-end reads de 300 pb.
- d: valor de la distancia entre read 1 y read 2
- r: tasa de mutaciones
- e: se establece en 0 para puntajes de calidad base de I, que es, equivalente a un Sanger Phred + 33 puntaje de 40.
- N: son los reads que queremos simular
- R: fracción de Indels
- X: probabilidad de extender un Indel

Por ejemplo:
wgsim -1300 -2300 -d500 -r0.001 -e0 -N100 -R0.10 -X0.30 A.\ muciniphila.fa Amuciphila.read1.fq Amuciphila.read2.fq

Utilizando este comando, vamos obteniendo los archivos fastq de cada una de las especies, como se aprecia en la imagen.

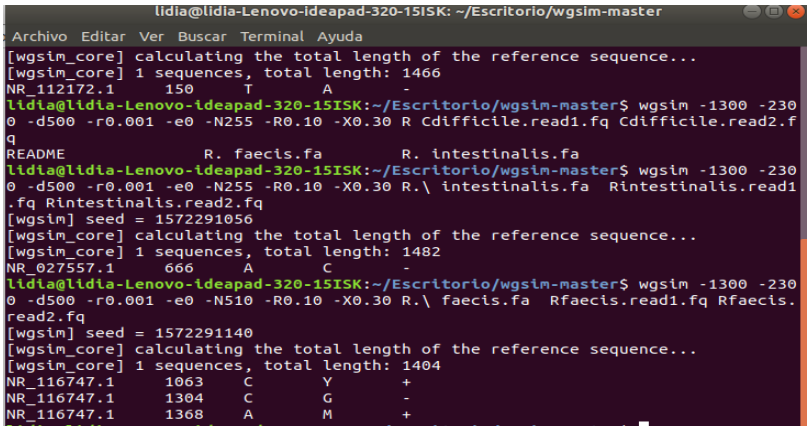

Figura 13: Análisis con Wgsim

2.4 Análisis bioinformático

En este apartado, vamos a ir definiendo los pasos seguidos en cada uno de los softwares que se han utilizado. Para evitar repeticiones y debido a que los análisis en cada uno de los softwares son largos, se mostrará el proceso completo para uno de los dataset.

2.4.1. QIIME2

Qiime (Quantitative Insights Into Microbial Ecology) es un software de código abierto escrito en Python y ejecutado en Linux. Ha sido desarrollado para realizar análisis de microbiomas (emplea el gen 16S o 18S) a partir de datos de secuenciación de ADN sin procesar, los cuales son generados por Illumina u otras plataformas. El software creado inicialmente fue Qiime1, sin embargo, debido al avance contínuo, este ha quedado en desuso en favor de Qiime2, todo el desarrollo y el soporte actualmente es únicamente para Qiime2 [13]. Qiime proporciona una herramienta web (https://view.qiime2.org/) en la cual podemos visualizar todos los artefactos tipo qzv, además cada uno de los artefactos proporcionados están compuestos por un fichero index.html que contiene la visualización de las mismas. Para ser utilizado, su descarga se realiza desde la página oficial de Qiime, a través de la terminal de Ubuntu. En la siguiente imagen podemos observar el diagrama de flujo que se sigue a la hora de trabajar con qiime2. Incluye demultiplexación, filtrado de calidad, selección de OTU, asignación taxonómica, reconstrucción filogenética, análisis y visualizaciones de diversidad.

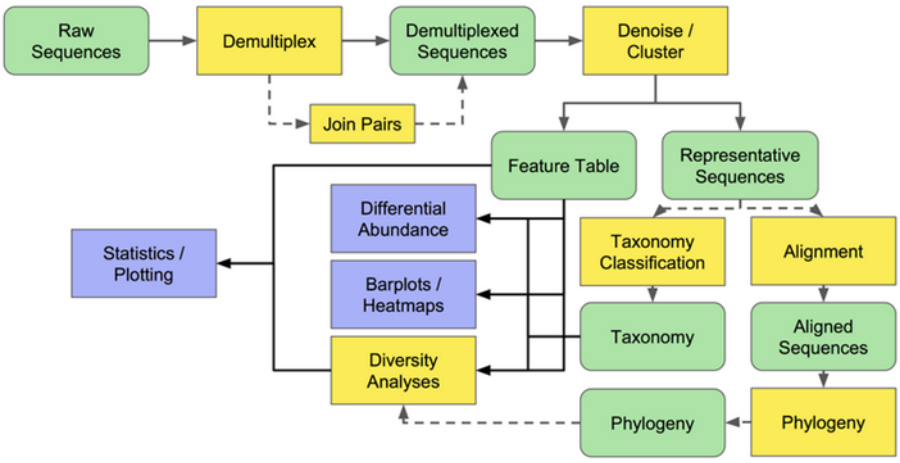

Figura 14: Flujo de trabajo en Qiime2

Los comandos utilizados para llevar a cabo el análisis están detallados en el anexo.

2.4.1.1 Análisis Microbiota intestinal:

El primer dataset analizado fue el de la microbiota intestinal. Para generar el análisis es necesario tener tanto los reads1 como los reads2 de cada una de las especies. Partíamos de 25 especies, por lo que tendremos 25 reads1 (forward) y 25 reads2 (reverse) en formato fastq. Otro archivo del cual tenemos que disponer es un sample-metadata.tsv. Los metadatos de qiime2 se almacenan normalmente en un archivo .tsv (en donde los valores están separados por tabulaciones). Estos archivos suelen tener una extensión de archivo.tsv o .txt Una vez realizado el sample-metadata.tsv se tiene que validar por Keemei. Este es un complemento de Excel que permite comprobar si cada una de las columnas generadas cumple con las especificaciones de Qiime2. En la siguiente imagen vemos como nuestro metadata cumple con dichas especificaciones. Una vez que tenemos los reads y el archivo metadata, el siguiente paso es importar las secuencias al software de Qiime2 para poder operar con ellas.

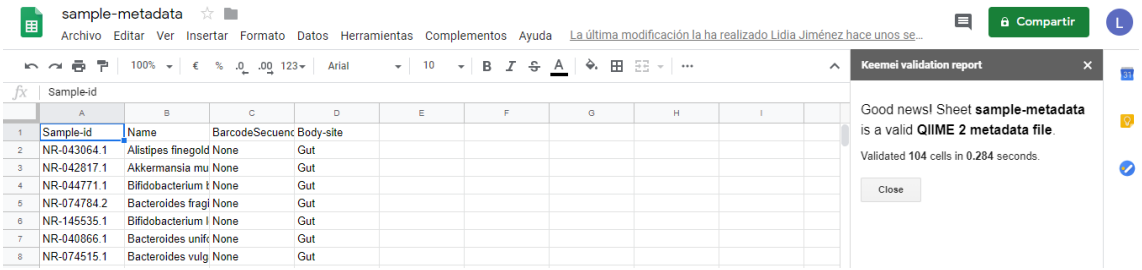

Figura 15: Archivo metadatos validado por Keemei

La importación de los datos se hizo a través del método Casava 1.8 paired-end demultiplexed fastq. Se nos devuelve un archivo demultiplexado. En el que, según la imagen siguiente, podemos apreciar un resumen de los reads y una gráfica que representa la frecuencia en función del número de reads.

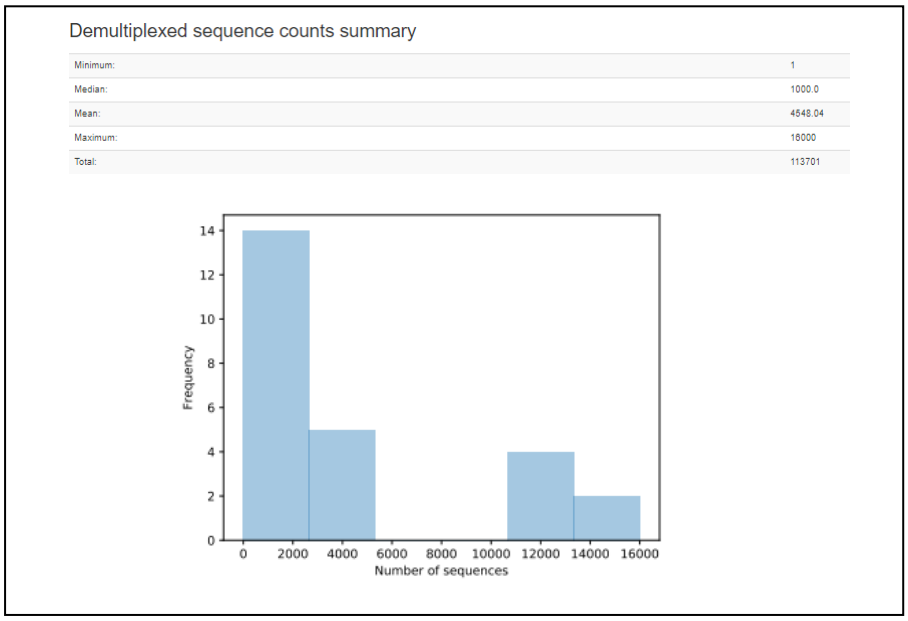

Figura 16: Artefacto del demux-paired-end.qzv

Seguidamente utilizamos el vsearch join-pairs para unir las secuencias. Al utilizar el qiime view podemos ver esta imagen. No tenemos caída de señal, todas las reads son de buena calidad, con un puntaje de calidad base de I, que es, equivalente a un Sanger Phred + 33 puntaje de 40.

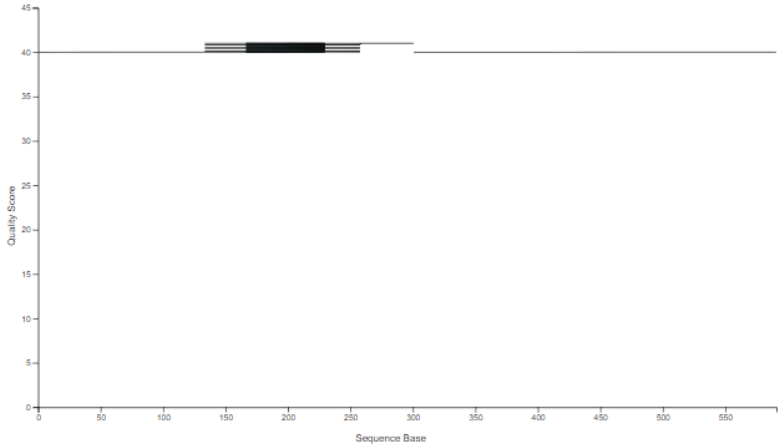

Figura 17: Calidad de las secuencias.

Al ser unas secuencias simuladas utilizaremos la opción dereplicate, cuya función es eliminar las secuencias que son duplicadas. Existen dos métodos para la eliminación de ruido: DADA2 y Deblur. Lo que hacen es discriminar los errores de la secuencia. En este trabajo no se utilizará ninguno de los dos, no existe ruido de fondo por lo que no se realizará ningún método de eliminación de ruido. Aún así, se trabajó con Deblur (tiene los tiempos de análisis más bajos que DADA2) y los comandos para su ejecución se muestran en el anexo también.
La asignación taxonómica es el paso más importante a llevar a cabo, se explora la composición taxonómica de las muestras y se relacionan con los metadatos. Se utilizará un algoritmo para clasificar y alinear las secuencias junto con una base de datos de referencia. Para llevar a cabo el análisis taxonómico de las secuencias, se utilizó la base de datos Greengenes. Se hará uso del clasificador Naive Bayes y el plugin gg-13-8-99-515-806-nb-classifier.qza. Aplicaremos este clasificador a nuestras secuencias y podremos generar una visualización.

El comando utilizado para llevar a cabo el análisis fue classify-sklearn. Se intentó llevar a cabo el análisis con classify-blast para que en su clasificación hiciera un barrido completo del genoma y no se quedara en una zona concreta, pero los tiempos de ejecución en la máquina virtual se excedieron a más de 72 horas, se perdía la conexión y no se obtenía ningún tipo de resultado. Por lo que finalmente se decidió utilizar el classificador Naive Bayes, aunque puede estar entrenando regiones V3-V4 del genoma 16S. No se dispone de información acerca de la localización de las regiones variables en el gen 16S rRNA de nuestras especies.
Una vez que obtenemos el artefacto taxa-bar-plots.qzv podemos ver el diagrama de barras de las especies. Lo podemos ver a cualquier nivel taxonómico, esta imagen corresponde a nivel de género. En donde cada columna de la imagen corresponde a una especie de la que partíamos representados frente a la frecuencia relativa. En esa misma página nos podemos descargar el formato csv de los niveles taxonómicos y así poder llevar a cabo el análisis de los resultados.

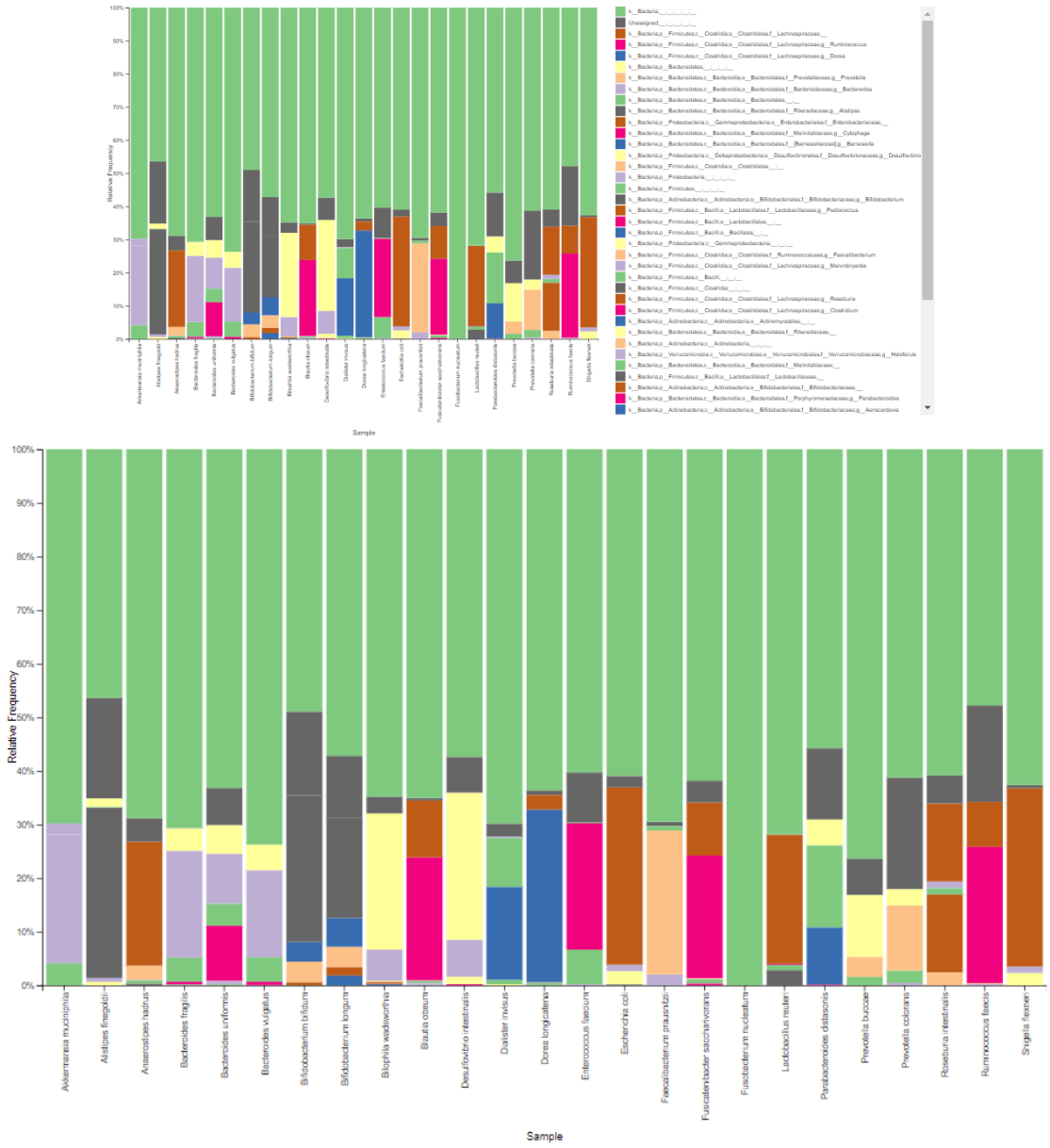

Figura 18: Representación gráfica de las asignaciones taxonómicas bacterianas en la microbiota intestinal correspondientes al taxón de género realizadas gracias al programa qiime2. La segunda imagen corresponde a la misma imagen más ampliada.

2.4.1.2 Análisis Microbiota Bucal:

En este caso, al igual que en el anterior análisis necesitamos los reads 1 y 2 y el archivo metadata. Tenemos 20 especies seleccionadas. Importamos los datos a qiime2 y se nos devuelve un archivo demultiplexado. Utilizando la herramienta de visualización de Qiime2, vemos un resumen de los reads y una gráfica que representa la frecuencia en función del número de reads por secuencia.

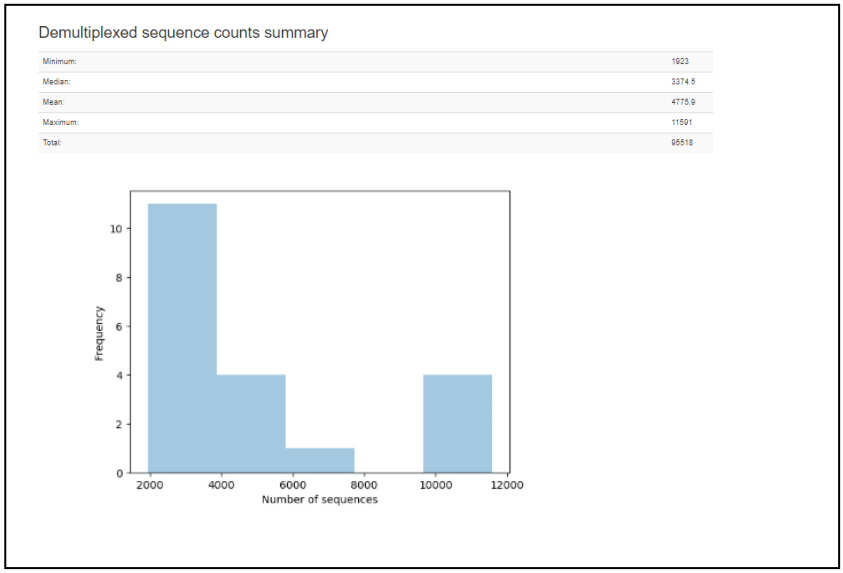

Figura 19: Artefacto del demux-paired-end.qzv Microbiota bucal

Al igual que en el dataset anterior tenemos una buena calidad de las secuencias. Por lo que no será necesario hacer ningún método para la eliminación de ruido.

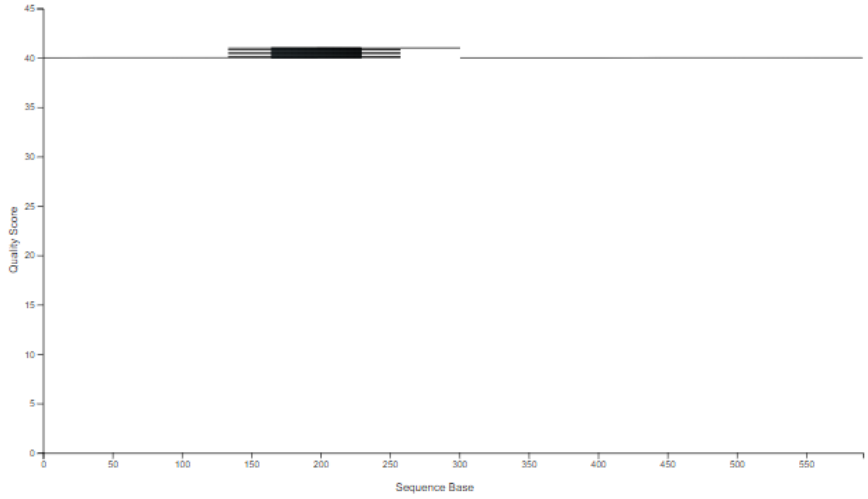

Figura 20: Calidad de las secuencias en la microbiota bucal

Haciendo la clasificación taxonómica de las especies nos encontramos con el siguiente diagrama de barras:

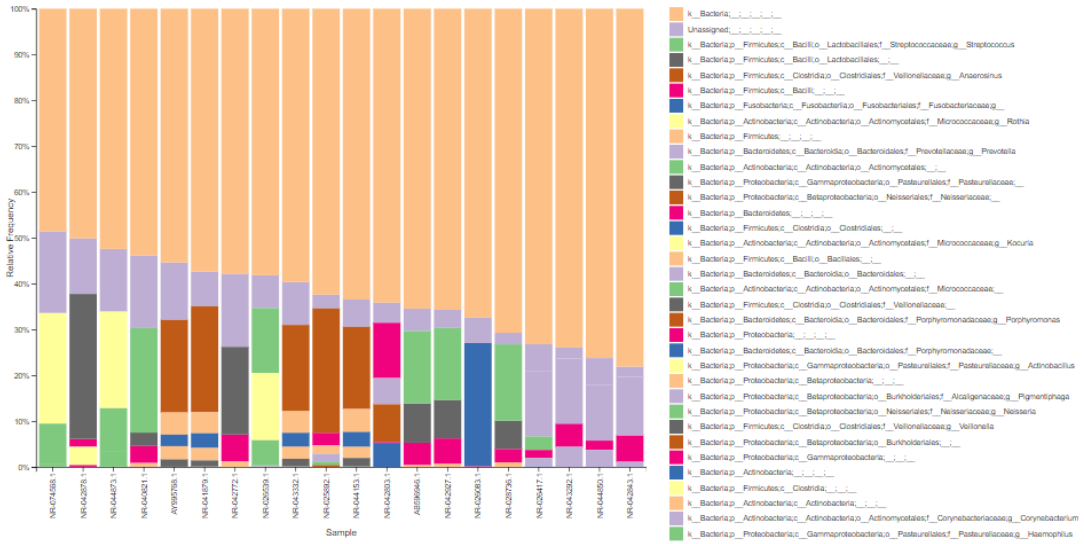

Figura 21: Representación gráfica de las asignaciones taxonómicas bacterianas en la microbiota bucal correspondientes al taxón de género realizadas gracias al programa qiime2

2.4.1.3 Análisis Microbiota Vaginal:

En este caso tenemos 19 especies a analizar. Se siguen los mismos pasos que en los datasets anteriores.

Demultiplexed sequence counts summary

Minimum:	500
Median:	5000.0
Mean:	5236.842105263157 5
Maximum:	14000
Total:	99500

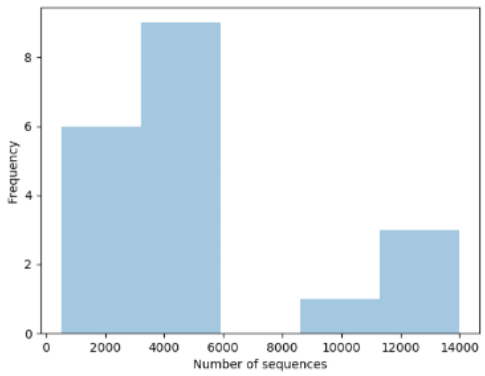

Figura 22: Artefacto del demux-paired-end.qzv Microbiota vaginal

En este caso, al igual que en los dataset anteriores existe una buena calidad de las secuencias. El análisis taxonómico se realiza de la misma manera que en dataset del microbioma intestinal.

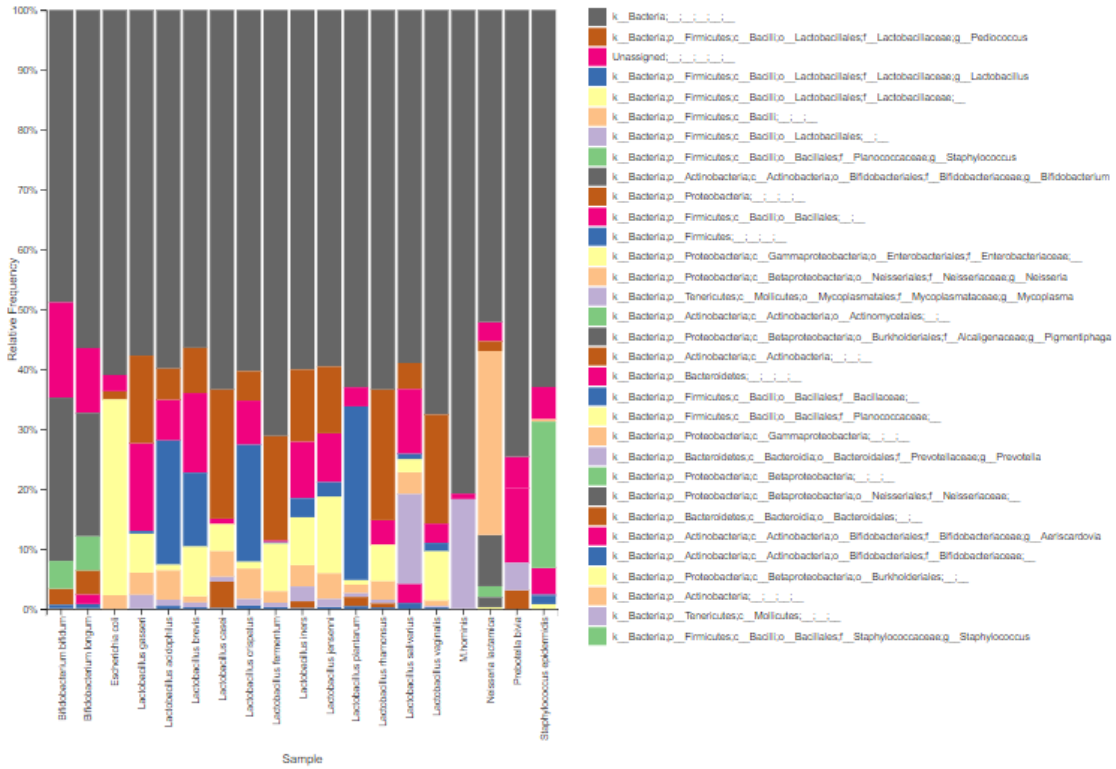

Figura 23: Representación gráfica de las asignaciones taxonómicas bacterianas en la microbiota vaginal correspondientes al taxón de género realizadas gracias al programa qiime2

2.4.2. Mothur

Mothur es un software de código abierto, escrito en C/C + +, este es un lenguaje compilado, lo que significa que una vez que se compila el código, no necesita otro programa para ejecutarlo. Mothur fue creado por el Dr. Patrick Schloss y su grupo de investigación en la Universidad de Michigan, con la finalidad de ser utilizado para el procesamiento de datos bioinformáticos. El software se ha descargado a través del repositorio de archivos GitHub, versión 1.43.0 [16]. A diferencia de otros softwares bioinformáticos, mothur puede ser utilizado en el sistema operativo de Windows. Al igual que Qiime2, procesa los datos a partir de una interfaz de línea de comandos, este no dispone de herramientas de visualización de datos, pero todos sus archivos de salida son archivos de texto.

El primer paso a seguir en Mothur es crear el entorno de trabajo, para ello, una vez que tenemos el programa instalado en la terminal de Ubuntu, es neceasario ejecutar: ./mothur y accedemos al programa.

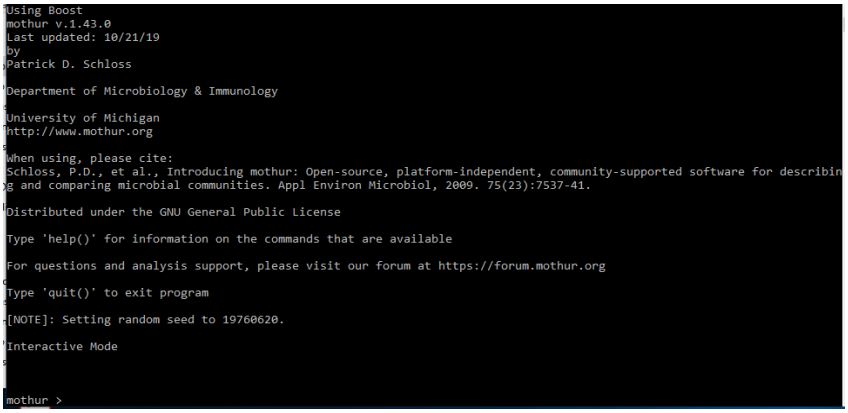

Figura 24: Entorno de trabajo de Mothur

2.4.2.1 Análisis Microbiota Bucal

Para llevar a cabo el análisis tenemos que importar los datos al software de mothur. Tenemos 20 especies de microbiota bucal, por lo que tendremos 20 reads 1 y 20 reads 2 para tratar.
Creamos dos ficheros de entrada y salida para almacenar los datos:
set.dir(input=./inputFiles)
set.dir(output=./outFiles)
En el inputFiles almacenaremos las reads, mientras que el outFiles, se almacenarán los ficheros que se vayan generando.

- Cargamos los ficheros

make.file(inputdir=/home/lidia/mothur-v.1.43.0/inputFiles/, type = fastq)
El comando make.file ordena los reads. Muestra un archivo en donde en cada línea se alinean los R1 y R2. La primera columna es el nombre de la muestra. La segunda columna es el nombre de la R1 para esa muestra y la tercera columna en el nombre de R2 para esa muestra.

- Ensamblado de las reads

Al tener las reads paired-end pareadas, la mejor forma de mejorar la calidad de la secuencia es ensamblando esas reads, utilizamos el siguiente comando:
make.contigs(file=/home/lidia/mothur-v.1.43.0/outFiles/stability.files, processors=2)
Desde ahora todas las reads están almacenadas en un archivo que es stability.trim.contings.fasta

- Resumen de resultados

Con el siguiente comando veremos el resumen del fichero fasta:
summary.seqs(fasta=stability.trim.contigs.fasta)

```
Using 2 processors.

                Start   End     NBases  Ambigs  Polymer NumSeqs
Minimum:        1       300     300     0       3       1
2.5%-tile:      1       402     402     0       4       2326
25%-tile:       1       466     466     0       4       23251
Median:         1       500     500     0       5       46501
75%-tile:       1       533     533     0       5       69751
97.5%-tile:     1       587     587     6       6       90676
Maximum:        1       600     600     103     7       93000
Mean:   1       498     498     0       4
# of Seqs:      93000

It took 1 secs to summarize 93000 sequences.
```

Figura 25: Resumen de resultados

En este fichero podemos ver las secuencias que fueron analizadas y el tamaño de las reads (nuestros reads deberían ser de 300pb). Si eliminamos las secuencias con más de 300pb y dejamos las que tienen al menos 100pb obtenemos:

screen.seqs(fasta=stability.trim.contigs.fasta, group=stability.contigs.groups, maxambig=0, maxlength=300, minlength=100)

```
Using 2 processors.

                Start   End     NBases  Ambigs  Polymer NumSeqs
Minimum:        1       300     300     0       4       1
2.5%-tile:      1       300     300     0       4       1
25%-tile:       1       300     300     0       4       1
Median:         1       300     300     0       4       2
75%-tile:       1       300     300     0       5       3
97.5%-tile:     1       300     300     0       4       3
Maximum:        1       300     300     0       5       3
Mean:   1       300     300     0       4
# of Seqs:      3

It took 0 secs to summarize 3 sequences.
```

Figura 26: Resumen screen.seqs

Sólo nos quedarían 3 secuencias para seguir con el proceso. Por lo que se decide cambiar las condiciones y poder seguir con el análisis. Modificamos el comando, tal que así:
screen.seqs(fasta=stability.trim.contigs.fasta, group=stability.contigs.groups, maxambig=0, maxlength=500)
Este comando elimina las secuencias con más de 500pb y elimina también las que contengan bases ambiguas(N). Ejecutamos el comando summary.seqs para comprobar los datos obtenidos:
summary.seqs(fasta=stability.trim.contigs.good.fasta)

```
                Start   End     NBases  Ambigs  Polymer NumSeqs
Minimum:        1       300     300     0       3       1
2.5%-tile:      1       387     387     0       4       967
25%-tile:       1       442     442     0       4       9669
Median:         1       466     466     0       5       19338
75%-tile:       1       484     484     0       5       29007
97.5%-tile:     1       499     499     0       6       37709
Maximum:        1       500     500     0       6       38675
Mean:   1       459     459     0       4
# of Seqs:      38675

It took 1 secs to summarize 38675 sequences.
```

Figura 27: Resumen de resultados

- Reducir el número de secuencias redundantes

Se hace uso del comando unique.seqs, lo que hace es combinar secuencias repetidas y el comando count.seqs para generar una tabla con los nombres de las secuencias en las filas y las columnas sería la cantidad de veces que aparece cada secuencia única en cada grupo.
unique.seqs(fasta=stability.trim.contigs.good.fasta)
count.seqs(name=stability.trim.contigs.good.names,group=stability.contigs.good.groups)

- Alineación de las secuencias

La alineación de las secuencias se realiza con el comando align.seqs. Utilizamos la base de datos Silva (versión 132) como referencia.
align.seqs(fasta=stability.trim.contigs.good.unique.fasta, reference=silva.nr_v132.align, flip=T)
Para la alineación, no hemos tomado ninguna región del gen 16S rRNA como referencia.
Resumen de la alineación:
summary.seqs(fasta=stability.trim.contigs.good.unique.align,
count=stability.trim.contigs.good.count_table)

```
Using 2 processors.

                Start   End     NBases  Ambigs  Polymer NumSeqs
Minimum:        1046    6411    300     0       3       1
2.5%-tile:      1106    10306   386     0       4       967
25%-tile:       5456    22113   441     0       4       9669
Median:         14287   32473   465     0       5       19338
75%-tile:       25298   40725   484     0       5       29007
97.5%-tile:     32804   43112   499     0       6       37709
Maximum:        37684   43116   500     0       6       38675
Mean:   15184   30616   459     0       4
# of unique seqs:       38251
total # of seqs:        38675

It took 22 secs to summarize 38675 sequences.
```

Figura 28: Resumen de resultados

- Volvemos a eliminar las secuencias mal alineadas.

screen.seqs(fasta=stability.trim.contigs.good.unique.align,count=stability.trim.contigs.good.count_table,summary=stability.trim.contigs.good.unique.summary, optimize=start-end-minlength, criteria=70)

- Filtramos con el comando filter.seqs

filter.seqs(fasta=stability.trim.contigs.good.unique.good.align, vertical=T, trump=.)
Eliminamos las secuencias que tengan huecos en su estructura.

- Volvemos a eliminar las secuencias duplicadas:

unique.seqs(fasta=stability.trim.contigs.good.unique.good.filter.fasta, count=stability.trim.contigs.good.good.count_table)

- pre.cluster(fasta=stability.trim.contigs.good.unique.good.filter.unique.fasta, count=stability.trim.contigs.good.unique.good.filter.count_table, diffs=2)

Utilizamos el comando pre.cluster para eliminar secuencias que son muy similares a las secuencias abundantes.

- Comando uchime

Este es un software incluido en Mothur, cuyo objetivo es eliminar las secuencias quiméricas.
chimera.uchime(fasta=stability.trim.contigs.good.unique.good.filter.unique.precluster.fasta, count=stability.trim.contigs.good.unique.good.filter.unique.precluster.count_table, dereplicate=t)

Con el comando remove.seq, utilizamos el archivo que nos devuelve UCHIME para eliminar esas secuencias quiméricas:
remove.seqs(fasta=stability.trim.contigs.good.unique.good.filter.unique.precluster.fasta, count=stability.trim.contigs.good.unique.good.filter.unique.precluster.count_table, accnos=stability.trim.contigs.good.unique.good.filter.unique.precluster.denovo.uchime.accnos)

- Clasificación de las secuencias:

Para la clasificación seguimos utilizando a Silva como base de datos.
classify.seqs(fasta=stability.trim.contigs.good.unique.good.filter.unique.precluster.pick.fasta, count=stability.trim.contigs.good.unique.good.filter.unique.precluster.denovo.uchime.pick.count_table, reference=silva.nr_v132.align, taxonomy=silva.nr_v132.tax, cutoff=80)

Con el comando classify.seqs las secuencias son emparejadas con una secuencia de referencia en la base de datos. Las secuencias que no son similares se elimina con el comando siguiente (remove.lineage)

- Eliminación de las secuencias que no han sido clasificadas.

remove.lineage(fasta=stability.trim.contigs.good.unique.good.filter.unique.precluster.pick.fasta, count=stability.trim.contigs.good.unique.good.filter.unique.precluster.denovo.uchime.pick.count_table, taxonomy=stability.trim.contigs.good.unique.good.filter.unique.precluster.pick.nr_v132.wang.taxonomy, taxon=unknown;)

- Generar OTUs

Usamos los siguientes comandos.
dist.seqs(fasta=stability.trim.contigs.good.unique.good.filter.unique.precluster.pick.fasta)
cluster.split(column=stability.trim.contigs.good.unique.good.filter.unique.precluster.pick.dist, count=stability.trim.contigs.good.unique.good.filter.unique.precluster.pick.count_table, method=average)

- Número de secuencias

Para saber el número de secuencias que hay en cada OTU utilizamos el comando make.shared, que contiene el conteo de OTU.
make.shared(list=stability.trim.contigs.good.unique.good.filter.unique.precluster.pick.an.list, count=stability.trim.contigs.good.unique.good.filter.unique.precluster.pick.count_table, label=0.03)

- Asignación Taxonómica

Se utiliza el comando classify.otu que toma la lista OTU y el resultado de count_table para asignar OTUs a taxones.

classify.otu(list=stability.trim.contigs.good.unique.good.filter.unique.precluster.pick.an.list, taxonomy=stability.trim.contigs.good.unique.good.filter.unique.precluster.pick.gg.wang.taxonomy, count=stability.trim.contigs.good.unique.good.filter.unique.precluster.pick.count_table, label=0.03, cutoff=80, basis=otu, probs=F)

- Construcción de la tabla OTU

Las tablas OTUs se representan por archivos .shared, este ha de convertirse a un archivo .biom de la siguiente forma:
make.biom(shared = stability.trim.contigs.good.unique.good.filter.unique.precluster.pick.an.0.03.subsample.shared, constaxonomy = stability.trim.contigs.good.unique.good.filter.unique.precluster.pick.an.0.03.cons.taxonomy)
De esta manera podrá convertirse posteriormente a un archivo de texto plano.

2.4.2.2 Análisis Microbiota Intestinal y Vaginal.

Para el análisis tanto de la microbiota intestinal como de la vaginal se siguieron los mismos comandos que para la microbiota bucal. Se omite para no ser repetitivo.

2.4.3. GAIA

Gaia es un software online desarrollado por la empresa Sequentia Biotech. Gaia nació de la necesidad de incorporar al mercado una herramienta que fuera capaz de identificar la diversidad biológica de un conjunto microbiano, debido a sus importantes aplicaciones en procesos industriales, medioambientales y clínicos. Y que a su vez fuera capaz de realizar análisis metagenómicos tanto por amplicón como por "shotgun" para organismos procariotas y eucariotas, permitiendo obtener una alta precisión y bajos tiempos de procesado. Gaia es capaz de analizar cualquier comunidad microbiana (bacterias y arqueas, virus y eucariotas) aplicadas a la genética humana, animal y vegetal [8]. Permite comparar muestras a tiempo real de cualquier nivel taxonómico.

2.4.3.1 Análisis Microbiota intestinal:
En primer lugar, se cargan los datos y se validan las muestras en función del secuenciador utilizado. Se muestran una serie de imágenes del proceso de análisis con Gaia.

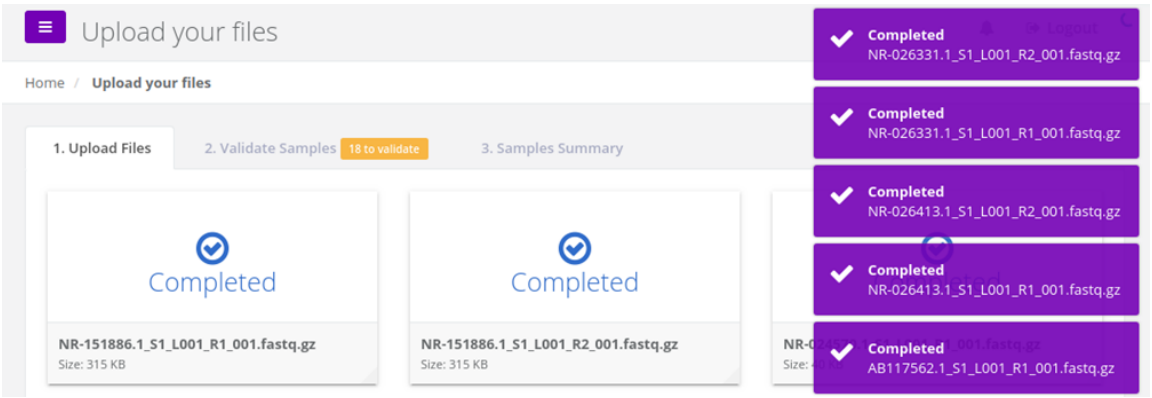

Figura 29: Subida de ficheros a Gaia

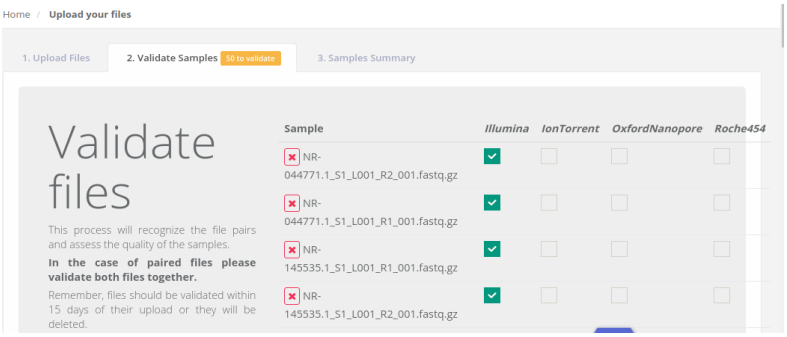

Figura 30: Validación de ficheros en Gaia

Una vez validadas las muestras, ya podemos crear el análisis. Este software tiene la ventaja de que una vez que termina con el proceso recibimos un e-mail automáticamente informando de que ya se ha realizado el análisis, lo cual elimina los tiempos muertos. El siguiente paso sería la identificación del tipo de análisis y la selección de la base de datos con la que queremos trabajar.

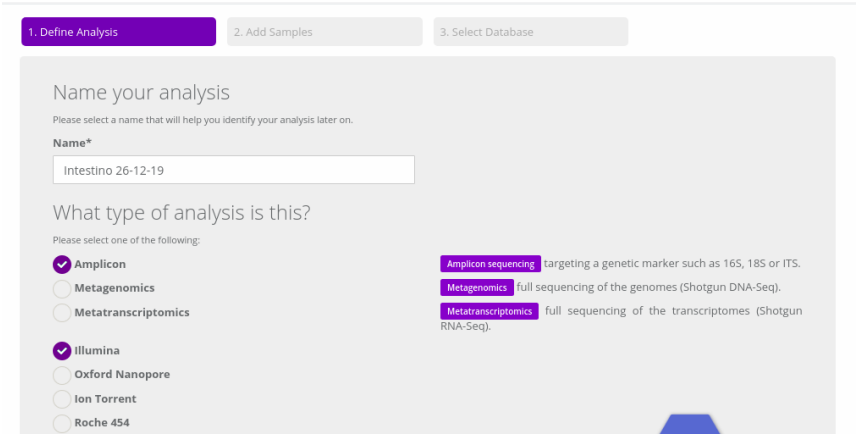

Figura 31: Selección del tipo de análisis

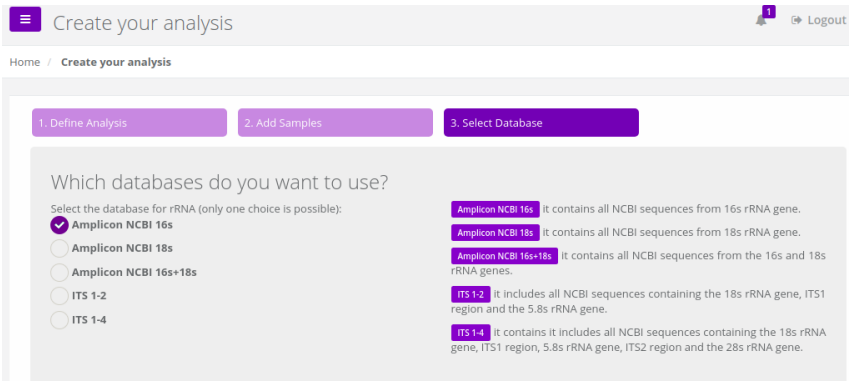

Figura 32: Elección de la base de datos

Una vez creado el análisis, en un tiempo de respuesta bastante pequeño podemos estar analizando datos. Una vez finalizado el proceso, se nos devuelve un archivo con un conjunto de datos en su interior. Estos ficheros nos ofrecen datos en relación a los distintos niveles taxonómicos, junto con análisis de diversidad alfa y beta. Además de ficheros de imágenes Krona, este es un visualizador interactivo usado en Metagenómica. La representación gráfica de la asignación taxonómica a nivel de género es la siguiente:

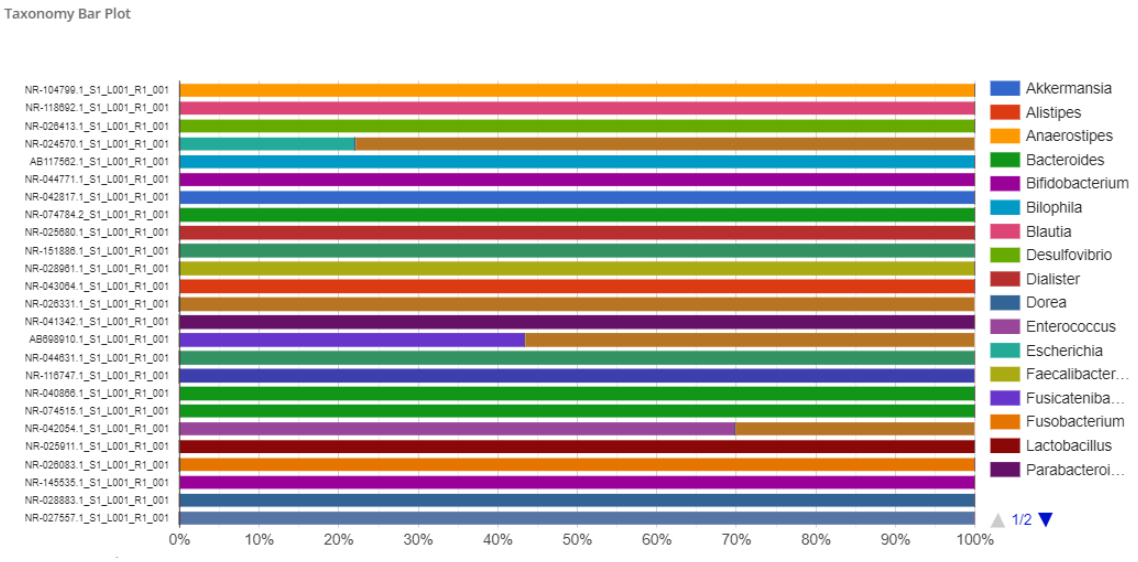

Figura 33: Representación gráfica de las asignaciones taxonómicas bacterianas en la microbiota intestinal correspondientes al taxón de género realizadas con el programa Gaia

2.4.3.2 Análisis Microbiota Bucal:

El proceso generado en el anterior dataset, es idéntico para este análisis. Únicamente se modifican los ficheros de partida. El diagrama de barras que se genera a nivel taxonómico es el siguiente:

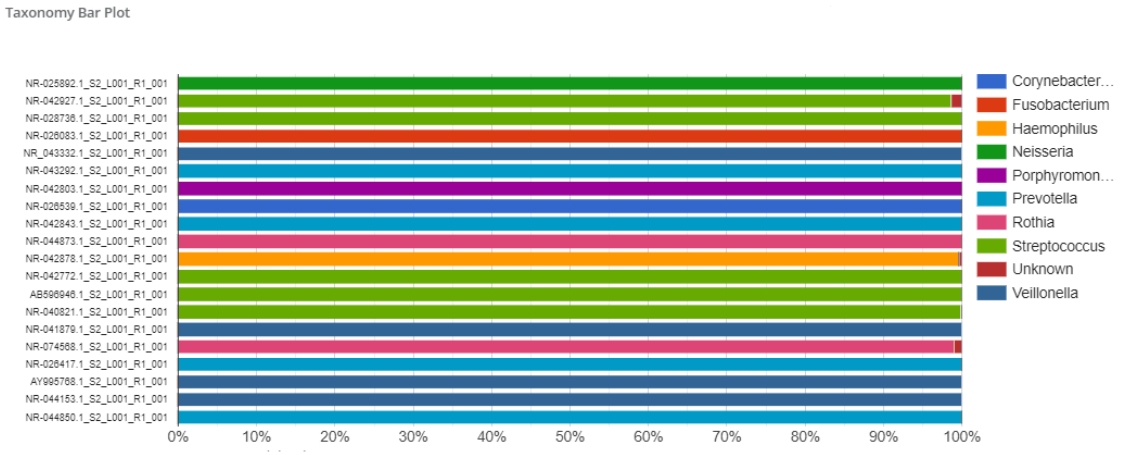

Figura 34: Representación gráfica de las asignaciones taxonómicas bacterianas en la microbiota bucal correspondientes al taxón de género realizadas con el programa Gaia

2.4.3.3 Análisis Microbiota Vaginal:

Pasos generados de la misma forma que en los casos anteriores.

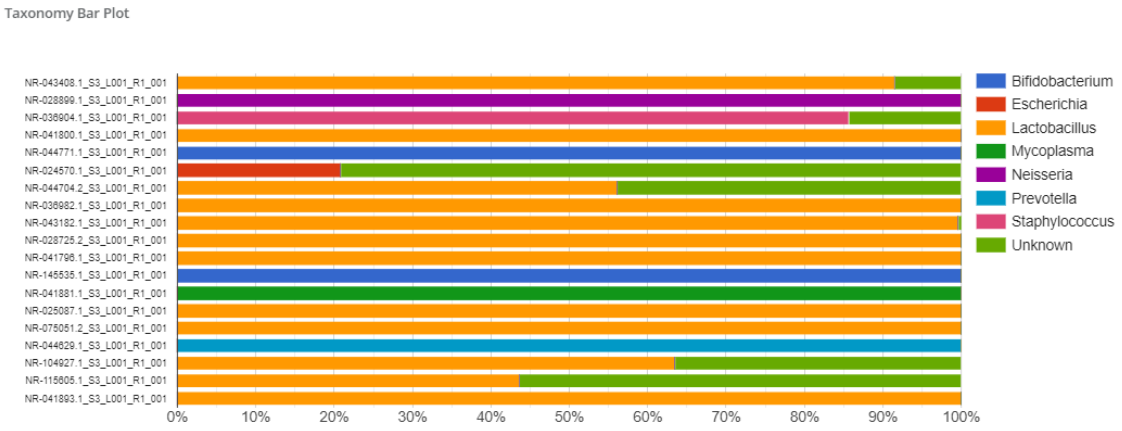

Figura 35: Representación gráfica de las asignaciones taxonómicas bacterianas en la microbiota vaginal correspondientes al taxón de género realizadas con el programa Gaia

3. Resultados

3.1 Control de calidad de los datos

Para hacer la comparativa de las herramientas, se han utilizado los siguientes parámetros: [8]
Precisión: Este valor indica como de bien el programa clasifica las reads.
Recall: Indica cómo de sensible es el programa a la hora de clasificar
F-measure o F-score: Corresponde a la medida armónica entre la precisión y el recall.

Estos parámetros fueron calculados de la siguiente manera:
Precisión: reads clasificados correctamente / reads clasificados
Recall= reads clasificados correctamente / (reads clasificados correctamente + reads No clasificados)
F-measure = 2 x ((precisión * recall) / (precisión + recall))

3.2 Análisis estadístico de los datos

Los datos han sido analizados taxonómicamente a nivel de género como ya se ha especificado. Se muestra una tabla con los valores obtenidos para los parámetros evaluados. Si nos fijamos en Qiime2, por ejemplo, podemos observar que, en cada uno de los parámetros, el valor asociado en los tres dataset es bastante similar. Al igual ocurre con los otros dos softwares, excepto en el caso de Mothur para el dataset 3, que no ha podido ser clasificado a nivel de género

	Dataset 1 (Intestinal)			Dataset 2 (Bucal)			Dataset 3 (Vaginal)		
	Precisión	Recall	F-measure	Precisión	Recall	F-measure	Precisión	Recall	F-measure
Qiime2	0,2963	0,7966	0,432	0,3053	0,7814	0,439	0,3361	0,7978	0,473
Mothur	0,1093	0,1999	0,1413	0,1569	0,736	0,2587	0	0	0
Gaia	0,8757	0,9231	0,8988	0,9284	0,9978	0,9619	0,8895	0,9127	0,9009

Tabla 6: Análisis de resultados

Si representamos gráficamente los resultados con el dataset intestinal, se comprueba esa gran diferencia entre los tres softwares. Gaia es el software que presenta los mejores resultados con respecto a Qiime2 y mothur. Se valorará su comportamiento en el apartado de conclusiones.

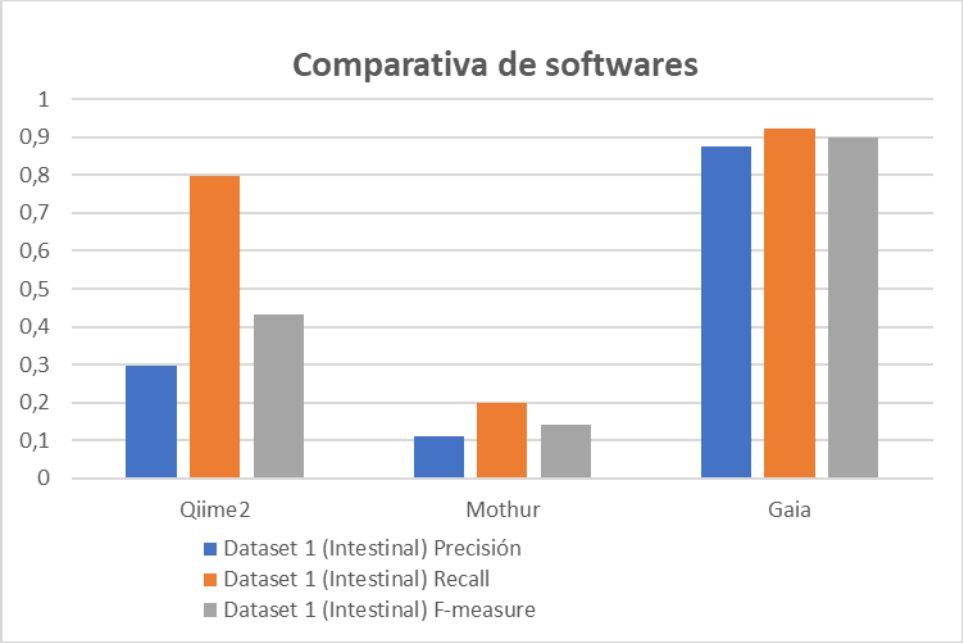

Figura 36: Comparativa de softwares

En esta otra figura representamos los tres datasets frente a los distintos parámetros evaluados en cada uno de los softwares utilizados. Al igual que en el caso anterior es Gaia quien ofrece mejores resultados.

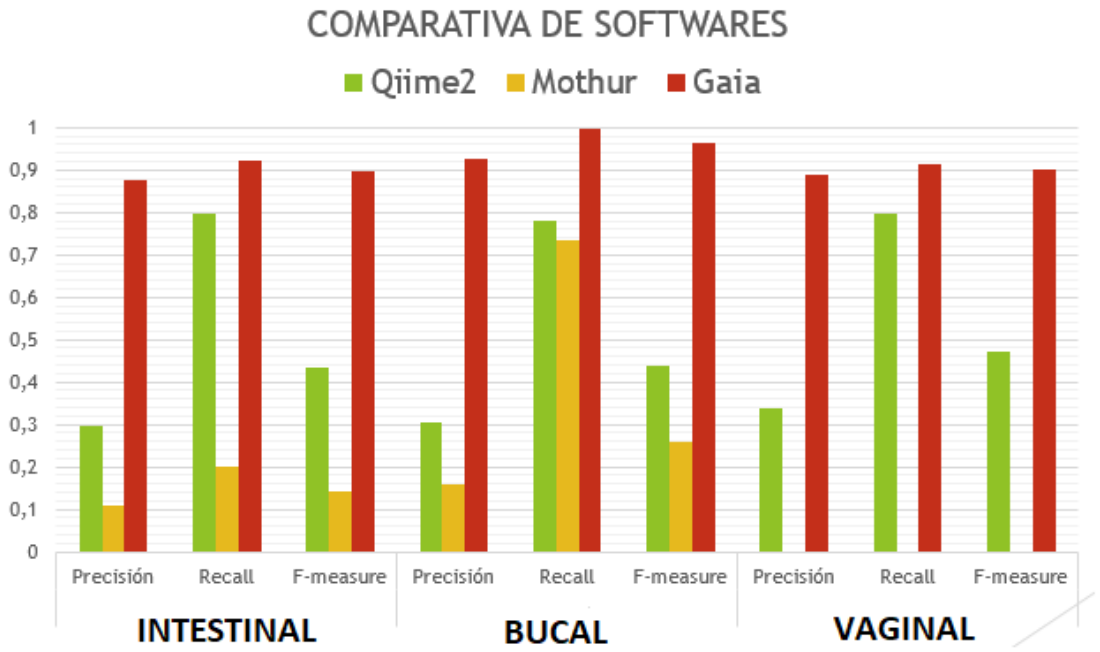

Figura 37: Comparativa de softwares en los tres datasets

4. Conclusiones

Como conclusión general, podemos decir que los datos obtenidos no son de confianza, se esperaba que con los tres softwares se obtuvieran resultados simulares debido a que son programas que son utilizados frecuentemente y sus funciones están bien establecidas, sin embargo, en nuestros resultados no se aprecia lo mismo. Se plantean una serie de valoraciones a raíz de los resultados obtenidos.

- Metodología y planificación

La planificación inicial no se planteó de la manera correcta, debido a que se dejó poco tiempo para el análisis con los diferentes softwares. Son herramientas densas, para las cuales se necesita mucho tiempo para poder comprenderlas en su totalidad.

- Simulación de reads

Habría que comprobar si ha ocurrido algún error a la hora de hacer la simulación con Wgsim e incluso a la hora de obtener las secuencias. Son muchos parámetros y las condiciones a tener en cuenta.

- Rendimiento Computacional

Inicialmente el rendimiento computacional fue muy bajo, hasta que finalmente se empezó a trabajar con Google Cloud no fue cuando se pudieron realizar los análisis con éxito.

- Análisis a nivel taxonómico

En todo momento se ha ejecutado el análisis para el nivel taxonómico de género. Hay que valorar la posibilidad de repetir el análisis para cualquier otro tipo de nivel, por ejemplo, orden o familia. En ese caso, es muy probable que los resultados mejoren para los tres softwares. No se ha podido realizar el análisis por falta de tiempo.

- Uso de softwares y clasificadores

Se ha trabajado mucho con el software de Qiime2, primero en la terminal de Ubuntu, notebook de Jupyter en Python y finalmente en la máquina virtual de Google Cloud. Se realizaron muchas pruebas para familiarizarnos con el programa. Otro de los errores de sus bajos resultados obtenidos radica en la clasificación realizada al utilizar el clasificador classify-sklearn. No se conocía la región en la que se realizaba el análisis, por lo que es muy probable que sólo mapeara la zona V3-V4 del gen 16S rRNA en vez de todas las regiones.

- Valoración de softwares

La herramienta GAIA cuenta con una interfaz gráfica e intuitiva y no requiere formación especializada para su uso, así que cualquier investigador que no sea experto en bioinformática la puede utilizar y evaluar fácilmente el análisis de los resultados obtenidos. Qiime2 ha sido bastante cómodo de utilizar gracias a su visualizador de resultados. Sin embargo, Mothur ha sido el software menos intuitivo a la hora de trabajar, al no disponer de interfaz gráfica es más complicado seguir los procesos de análisis, aunque en todo momento se pueden ir visualizando en formato de texto plano.

Aún teniendo malos resultados, el balance final es positivo. Se ha trabajado un tema y unos softwares totalmente desconocidos y se ha llegado a comprender el proceso de análisis metagenómico.

5. Glosario

Amplicón: segmento de DNA o RNA que resulta de la amplificación (producción de múltiples copias) de una secuencia de interés.
C / C + +: lenguaje de programación.
Conting: Conjunto de fragmentos solapantes de los que se pueden obtener una secuencia más larga.
Dataset: Conjunto de datos utilizados para llevar a cabo el análisis.
Delección: tipo de alteración genética que consiste en la pérdida de un fragmento de DNA dentro de un cromosoma.
Disbiosis: Cambios en la configuración estructural y/o funcional de la microbiota intestinal que producen alteraciones en la homeóstasis del huésped, favoreciendo la susceptibilidad a ciertas enfermedades.
In Silico: expresión que significa que se ha realizado por simulación computacional.
INDEL: alteración genética en la cual se produce una inserción o una deleción.
Inserción: tipo de alteración genética que consiste en la ganancia de un fragmento de DNA dentro de un cromosoma.
Máquina virtual: es un software que simula otro dispositivo, de tal modo que puedes ejecutar otro sistema operativo en su interior. Todos sus componentes son virtuales.
Read: cada una de las secuencias que se generan durante el proceso de secuenciación.
OTU: Unidad taxonómica operativa.
NGS: Next Generation Sequencing.
Python: lenguaje de programación.
rRNA: ribosomal RNA.

6. Bibliografía

[1] Emanuele Rinninella el al., “What is the Healthy Gut Microbiota Composition? A Changing Ecosystem across Age, Environment, Diet, and Diseases”, *microorganisms*,2019,7,14.
[2] Rodicio MR, Mendoza MC. Identificación bacteriana mediante secuenciación del ARNr16S: fundamento, metodología y aplicaciones en microbiología clínica. Enferm Infecc Microbiol Clin 2004; 22(4): 238-45.
[3] A. Suárez Moya, Microbioma y secuenciación masiva.
[4] RStudio Team (2015) RStudio: Integrated Development for R. RStudio, Inc., Boston, MA. http://www.rstudio.com/
[5] Li H. wgsim - Read simulator for next generation sequencing. Github Repository. 2011 [online] http://github.com/lh3/wgsim.
[6] https://en.m.wikipedia.org/wiki/FASTQ_format; 03/11/2019
[7] Perrine, H., Lagier, J.-C., Colson, P., Bittar, F., & Raoult, D. (2017). Repertoire of human gut microbes. Microbial Pathogenesis, 106,103 112.
[8] GAIA: an integrated metagenomics suite. Paytuví A., Battista E., Scippacercola F., Aiese Cigliano R., Sanseverino W. Sequentia Biotech SL, Calle Comte d’Urgell 240, 08036 Barcelona, Spain.
[9] https://en.wikipedia.org/wiki/FASTA_format; 03/11/2019
[10] https://es.wikipedia.org/wiki/Illumina_(compa%C3%B1%C3%ADa);03/11/2019
[11] An introduction to Next-Generation Sequencing Technology. Illumina,Inc.
[12] https://en.m.wikipedia.org/wiki/Phred_quality_score; 04/11/2019
[13] Bolyen E, Rideout JR, Dillon MR, Bokulich NA, Abnet CC, Al-Ghalith GA, Alexander H, Alm EJ, Arumugam M, Asnicar F, Bai Y, Bisanz JE, Bittinger K, Brejnrod A, Brislawn CJ, Brown CT, Callahan BJ, Caraballo-Rodríguez AM, Chase J, Cope EK, Da Silva R, Diener C, Dorrestein PC, Douglas GM, Durall DM, Duvallet C, Edwardson CF, Ernst M, Estaki M, Fouquier J, Gauglitz JM, Gibbons SM, Gibson DL, Gonzalez A, Gorlick K, Guo J, Hillmann B, Holmes S, Holste H, Huttenhower C, Huttley GA, Janssen S, Jarmusch AK, Jiang L, Kaehler BD, Kang KB, Keefe CR, Keim P, Kelley ST, Knights D, Koester I, Kosciolek T, Kreps J, Langille MGI, Lee J, Ley R, Liu YX, Loftfield E, Lozupone C, Maher M, Marotz C, Martin BD, McDonald D, McIver LJ, Melnik AV, Metcalf JL, Morgan SC, Morton JT, Naimey AT, Navas-Molina JA, Nothias LF, Orchanian SB, Pearson T, Peoples SL, Petras D, Preuss ML, Pruesse E, Rasmussen LB, Rivers A, Robeson MS, Rosenthal P, Segata N, Shaffer M, Shiffer A, Sinha R, Song SJ, Spear JR, Swafford AD, Thompson LR, Torres PJ, Trinh P, Tripathi A, Turnbaugh PJ, Ul-Hasan S, van der Hooft JJJ, Vargas F, Vázquez-Baeza Y, Vogtmann E, von Hippel M, Walters W, Wan Y, Wang M, Warren J, Weber KC, Williamson CHD, Willis AD, Xu ZZ, Zaneveld JR, Zhang Y, Zhu Q, Knight R, and Caporaso JG. 2019. Reproducible, interactive, scalable and extensible microbiome data science using QIIME 2. Nature Biotechnology 37: 852–857. https://doi.org/10.1038/s41587-019-0209-9
[14] Roxana J. Hickey, Xia Zhou, Jacob D. Piersonb, Jacques Ravelc, and Larry J.Forneya. Understanding vaginal microbiome complexity from an ecological perspective. Transl Res. 2012 October; 160(4): 267–282. doi:10.1016/ j.trsl .2012.02.008.
[15] Héctor Alejandro Serrano-Coll, Miryan Sánchez-Jiménez, Nora Cardona-Castro. Conocimiento de la microbiota de la cavidad oral a través de la metagenómica. Revista CES Odontología ISSN 0120-971X. Volumen 28 No. 2 Segundo Semestre de 2015
[16] Schloss PD et al. (2009). Introducing mothur: open-source, platform-independent, communitysupported software for describing and comparing microbial communities. *Applied and Environmental Microbiology*, 75: 7537-7541.
[17] Thomas J. Sharpton, An introduction to the analysis of shotgun metagenomic data. Front Plant Sci. 2014; 5: 209.
[18] Florian P. Breitwieser, Jennifer Lu and Steven L. Salzberg. A review of methods and databases for metagenomic classification and assembly. Briefings in Bioinformatics, 2017, 1–15

[19] Alexandre Almeida, Alex L. Mitchell, Aleksandra Tarkowska and Robert D. Finn, Benchmarking taxonomic assignments base don 16S rRNA gene profiling of the microbiota, from commonly sampled environments, GigaScience,7,2018,1-10.
[20] Bioinformatics Analysis of Massive Parallel Sequencing, Pablo Marin-Garcia

7. Anexos

7.1 Anexo I: Comandos utilizados en QIIME2

El primer paso es crear el entorno de qiime2 ejecutando: source actívate qiime2-2019.10

- Importación de datos:

```
qiime tools import \
  --type 'SampleData[PairedEndSequencesWithQuality]' \
  --input-path casava-18-paired-end-demultiplexed \
  --input-format CasavaOneEightSingleLanePerSampleDirFmt \
  --output-path demux-paired-end.qza
```

- Visualización de la demultiplexación:

```
qiime demux summarize \
  --i-data demux-paired-end.qza \
  --o-visualization paired-end-demux.qzv

mv demux-paired-end.qza demux.qza
```

- Unión de lecturas con vsearch:

```
qiime vsearch join-pairs \
  --i-demultiplexed-seqs demux.qza \
  --o-joined-sequences demux-joined.qza
```

- Eliminación de duplicados:

```
qiime vsearch dereplicate-sequences \
--i-sequences demux-joined.qza \
--o-dereplicated-table table \
--o-dereplicated-sequences rep-seqs
```

- Para obtener table.qzv y rep-seqs.qzv

```
qiime feature-table summarize \
  --i-table table.qza \
  --o-visualization table.qzv \
  --m-sample-metadata-file sample-metadata.tsv

qiime feature-table tabulate-seqs \
  --i-data rep-seqs.qza \
  --o-visualization rep-seqs.qzv
```

- Análisis Taxonómico

```
qiime feature-classifier classify-sklearn \
  --i-classifier gg-13-8-99-515-806-nb-classifier.qza \
  --i-reads rep-seqs.qza \
  --o-classification taxonomy.qza

qiime taxa barplot \
  --i-table table.qza \
  --i-taxonomy taxonomy.qza \
  --m-metadata-file sample-metadata.tsv \
  --o-visualization taxa-bar-plots.qzv
```

- Opción Deblur:

En el caso de que se quiera utilitzar la opción deblur, se partiria desde el archivo demux-paired-end.qza

```
qiime quality-filter q-score \
 --i-demux demux-paired-end.qza \
 --o-filtered-sequences demux-filtered.qza \
 --o-filter-stats demux-filter-stats.qza

qiime deblur denoise-16S \
  --i-demultiplexed-seqs demux-filtered.qza \
  --p-trim-length -- \
  --o-representative-sequences rep-seqs-deblur.qza \
  --o-table table-deblur.qza \
  --p-sample-stats \
  --o-stats deblur-stats.qza

qiime metadata tabulate \
  --m-input-file demux-filter-stats.qza \
  --o-visualization demux-filter-stats.qzv

qiime deblur visualize-stats \
  --i-deblur-stats deblur-stats.qza \
  --o-visualization deblur-stats.qzv
```

(renombrar)

```
mv rep-seqs-deblur.qza rep-seqs.qza
mv table-deblur.qza table.qza
```

(volver a Análisis Taxonómico, en el caso de que se haya hecho Deblur)

- Generar un árbol para análisis de diversidad filogenética

```
qiime phylogeny align-to-tree-mafft-fasttree \
  --i-sequences rep-seqs.qza \
  --o-alignment aligned-rep-seqs.qza \
  --o-masked-alignment masked-aligned-rep-seqs.qza \
  --o-tree unrooted-tree.qza \
  --o-rooted-tree rooted-tree.qza
```

- Análisis diversidad alfa y beta:

```
qiime diversity core-metrics-phylogenetic \
  --i-phylogeny rooted-tree.qza \
  --i-table table.qza \
  --p-sampling-depth 15 \
  --m-metadata-file sample-metadata.tsv \
  --output-dir core-metrics-results

qiime diversity alpha-group-significance \
  --i-alpha-diversity core-metrics-results/faith_pd_vector.qza \
  --m-metadata-file sample-metadata.tsv \
  --o-visualization core-metrics-results/faith-pd-group-significance.qzv

qiime diversity alpha-group-significance \
  --i-alpha-diversity core-metrics-results/evenness_vector.qza \
```

```
  --m-metadata-file sample-metadata.tsv \
  --o-visualization core-metrics-results/evenness-group-significance.qzv

qiime diversity beta-group-significance \
  --i-distance-matrix core-metrics-results/unweighted_unifrac_distance_matrix.qza \
  --m-metadata-file sample-metadata.tsv \
  --m-metadata-column Body-site \
  --o-visualization core-metrics-results/unweighted-unifrac-body-site-significance.qzv \
  --p-pairwise

qiime diversity beta-group-significance \
  --i-distance-matrix core-metrics-results/unweighted_unifrac_distance_matrix.qza \
  --m-metadata-file sample-metadata.tsv \
  --m-metadata-column subject \
  --o-visualization core-metrics-results/unweighted-unifrac-subject-group-significance.qzv \
  --p-pairwise
```

8. Agradecimientos

En primer lugar, me gustaría agradecer a mi tutor, Dr. Andreu Paytuví, por la confianza depositada en mi para llevar a cabo este proyecto, por su dedicación y supervisión. Agradecer a todos aquellos que me han animado y apoyado durante estos dos largos años de Máster, un trabajo duro en donde a veces no veía la luz al final del túnel. Por último, agradecer a las personas más importantes en mi vida, a mis padres, mis hermanos, mis sobrinos y a mi pareja. Sin ellos no estaría aquí.

Muchas gracias.