INFORME TRANSCRIPTÓMICA

María Legarreta Uriarte
26/03/2024

APARTADO 1

Para la realización de este apartado se realizó el siguiente workflow:

1. Control de calidad utilizando FastQC
2. Alineamiento de las secuencias utilizando HISAT2
3. Conteo de las lecturas con HTSEQ

Los comandos utilizados, así como las versiones de los programas utilizados, se detallarán más adelante según se vaya avanzando en el desarrollo del trabajo. Para la realización de este apartado se utilizó un ambiente denominado transcriptómica (presente en el directorio envs proporcionado).

Todos los comandos utilizados para este primer apartado fueron lanzados desde dentro del directorio "ejercicio_final/Apartado1"

CONTROL DE CALIDAD FASTQC

Para el control de calidad de las muestras proporcionadas se utilizó el programa FastQC (versión 0.12.1). Para ello, se introdujo el comando `fastqc` en la consola de bash, con lo que se abrió el programa FastQC. En el apartado "File>Open" se introdujeron los cuatro archivos .fastq proporcionados y se realizaron informes de cada uno de ellos (seleccionando "File>Save report"). El programa genera dos tipos de archivos para cada uno de los archivos .fastq proporcionados:

- Un archivo .html que redirige al usuario a una dirección web donde se puede analizar cada uno de los diagramas para cada secuencia.
- Un archivo .zip que contiene la misma información que el archivo .html.

Los archivos .html y .zip para cada una de las secuencias se encuentran en el directorio Apartado1/resultados/fastqc. A continuación se muestran algunos de los datos y gráficas más reseñables.

Estadísticas generales

En la **Tabla 1** se muestra la información más importante de cada una de las secuencias analizadas.

Tabla 1. Resumen datos estadísticos.

Muestra	Codificación Ilumina	Secuencias totales	Bases totales	Longitud secuencia	%GC
SRR479052.chr21_1	1.9	15340	1.5 Mpb	101	52
SRR479052.chr21_2	1.9	15340	1.5 Mpb	101	52
SRR479054.chr21_1	1.9	9746	984.3 kpb	101	51
SRR479054.chr21_2	1.9	9746	984.3 kpb	101	51

Los diferentes apartados indican lo siguiente:

- **Muestra:** indica el nombre de la muestra, así como la direccionalidad de las lecturas. Los archivos _1 tienen direccionalidad 5’ - 3’ y los archivos _2 tienen direccionalidad 3’ - 5’.
- **Codificación Ilumina:** Indica la codificación de la calidad de las bases.
- **Secuencias totales:** Número de lecturas.
- **Bases totales:** Número de bases.
- **Longitud de la secuencia:** Longitud de las lecturas
- **%GC:** Contenido de GC presente en cada una de las lecturas.

Calidad de la secuencia por base

En la **Figura 1** se muestran las gráficas que indican los resultados del análisis por base de cada una de las muestras y secuencias.

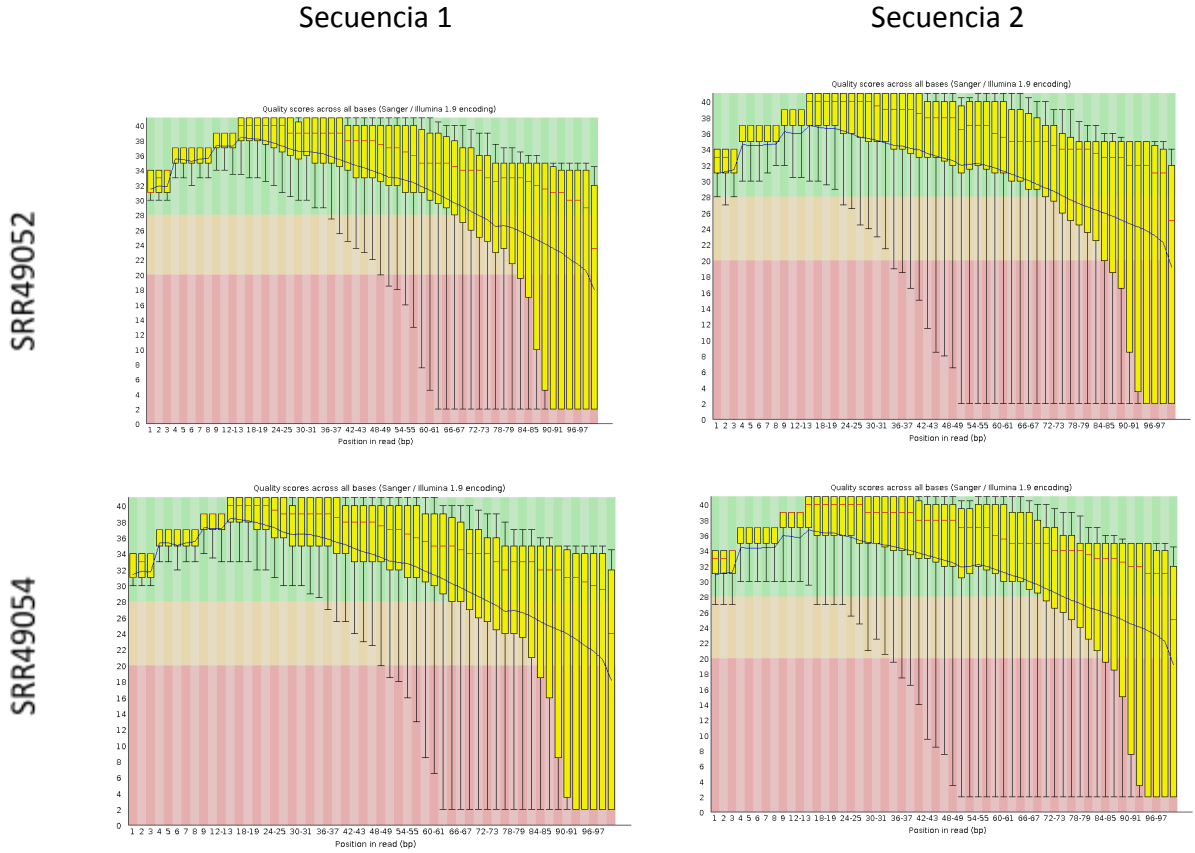

Figura 1. Diagramas de calidad de las secuencias por base. La calidad se representa en escala Phred.

Se puede observar que la calidad de las secuencias empieza a dejar de ser óptima a partir de la posición 67, y pasa a ser de mala calidad a partir de la posición 84 aproximadamente para las cuatro muestras.

Esto indica que para un correcto alineamiento sería necesario recortar esas regiones de las muestras. Sin embargo, este trabajo se ha realizado sin dicha modificación.

Quality scores por secuencia

El per sequence quality score es una medida que indica la calidad de cada base en todas las secuencias generadas por un proceso de secuenciación (**Figura 2**). Esta medida es fundamental para evaluar la confiabilidad de los datos de secuenciación, ya que permite identificar posibles

problemas como artefactos de secuenciación, errores sistemáticos o regiones de baja calidad en las secuencias.

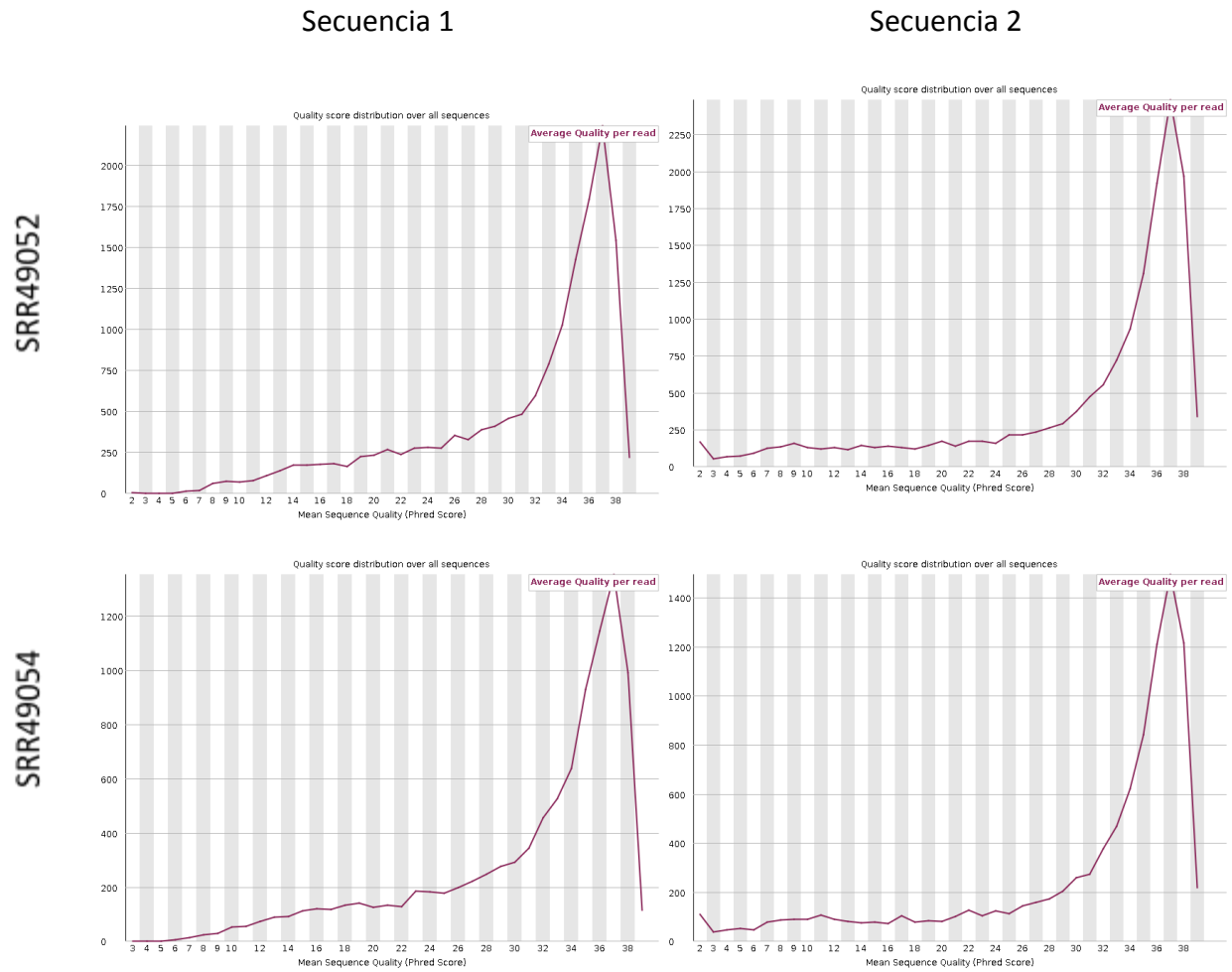

Figura 2. Quality scores de cada una de las muestras.

En las gráficas presentadas se destaca un pico significativo en el Phred Score, aproximadamente en 36, lo que sugiere una calidad promedio de lectura en torno a este valor. Este hallazgo es altamente alentador, ya que un Phred Score de 36 indica una calidad muy buena de las secuencias obtenidas.

Contenido de la secuencia por base

El contenido de secuencia por base (per base sequence content) es un análisis que examina la distribución de nucleótidos en cada posición de las lecturas secuenciadas (**Figura 3**). Esta

métrica es crucial para detectar posibles sesgos en la composición de bases a lo largo de las secuencias, lo que podría indicar problemas durante la preparación de la muestra o la secuenciación misma, como la presencia de adaptadores o contaminantes. Una distribución uniforme y equilibrada de los nucleótidos a lo largo de las lecturas es deseable para garantizar una calidad óptima de los datos y una interpretación precisa de los resultados del análisis genómico.

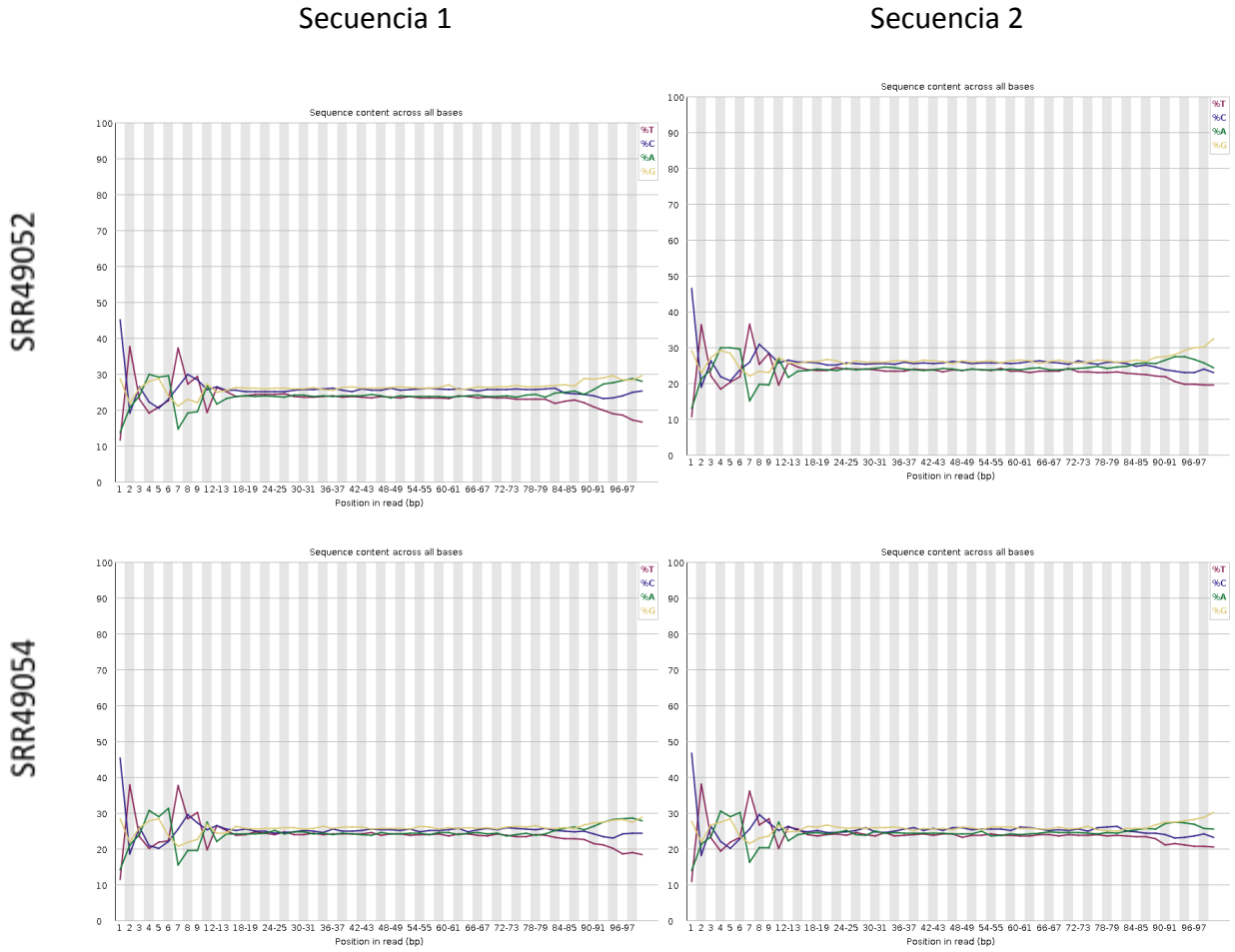

Figura 3. Contenido de la secuencia por base en cada una de las muestras.

Como se puede apreciar, las gráficas correspondientes a este apartado muestran errores en las cuatro muestras, lo cual es coherente con la técnica de RNASeq. Esto se debe al uso de "random hexamer primers" durante la generación de cDNA de las librerías, una práctica común que puede resultar en cierta variabilidad en la calidad de las secuencias obtenidas.

Secuencias sobre-representadas

FastQC es capaz de detectar secuencias que están sobre-representadas en el conjunto de datos, lo cual puede sugerir la presencia de artefactos o contaminantes. Estas secuencias pueden venir de adaptadores, secuencias de vectores, o incluso secuencias biológicas que se amplifican de manera desproporcionada durante la preparación de la muestra. En el caso de las muestras proporcionadas, se muestran en la **Tabla 2** las secuencias sobre-representadas para cada una de las muestras.

Tabla 2. Secuencias sobre-representadas por muestra.

Muestra	Secuencia	Conteo	Porcentaje (%)
SRR479052.chr21_1	CTTTTACTTCCTCTAGATAGTCAAGTTCGACCGTCTTCTCAGCGCTCCGC	21	13.69
SRR479052.chr21_2	CTAACACGTGCGCGAGTCGGGGGCTCGCACGAAAGCCGCCGTGGCGCAAT	20	13.04

Como se puede apreciar, solamente se han encontrado secuencias sobre-representadas en la muestra SRR479052, en ambas secuencias.

Contenido de adaptadores

El apartado "Adapter Content" en FastQC evalúa la presencia de secuencias de adaptadores en el conjunto de datos de secuenciación. Los adaptadores son secuencias cortas de ADN que se utilizan durante la preparación de las muestras para la secuenciación, pero que no forman parte del material genético de interés (**Figura 4**).

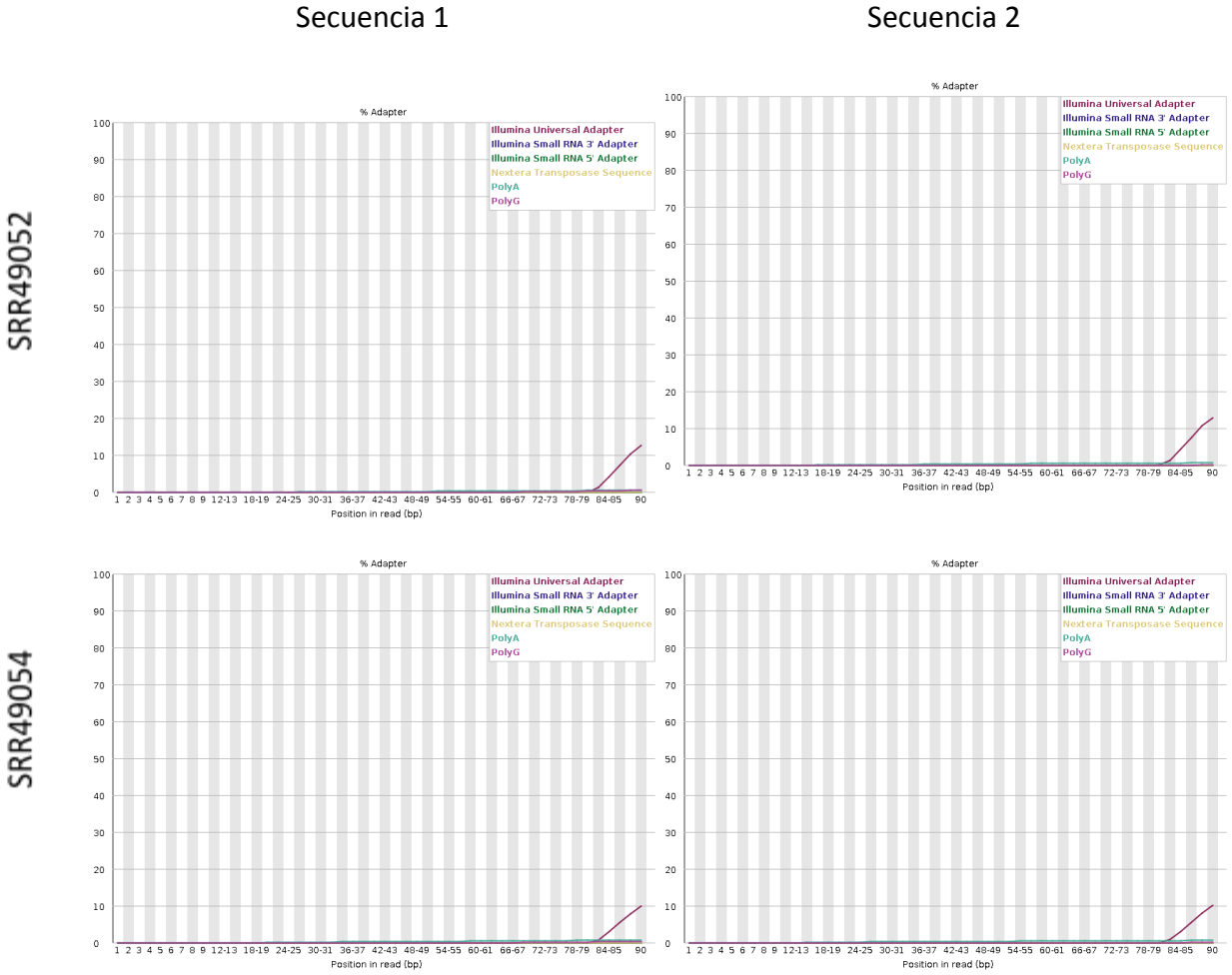

Figura 4. Contenido de adaptadores en cada una de las muestras.

Se observa la presencia de adaptadores universales de Illumina en todas las muestras, detectándose desde aproximadamente 80 pares de bases (pb) hasta un 10% de la longitud total de la secuencia.

ALINEAMIENTO E INDEXADO DEL GENOMA

Para el alineamiento de las secuencias se utilizó el software HISAT2 (versión 2.2.1). HISAT2 es un alineador de secuencias de ARN de alto rendimiento que utiliza un enfoque basado en grafos para mapear de manera eficiente las lecturas de ARN a un genoma de referencia. Utiliza índices de grafos de Burrows-Wheeler (BWT) compactos y sensibles para realizar alineaciones precisas incluso en regiones genómicas altamente variables, como los puntos de empalme. HISAT2 también es capaz de alinear lecturas de ARN splice-alineadas, lo que significa que puede identificar correctamente las uniones entre exones e intrones en el ARN mensajero.

Indexado del genoma

Antes de realizar el alineamiento, es necesario indexar el genoma si no se ha realizado ya para que el alineamiento se dé de forma más rápida y efectiva. Para este paso se utilizó el siguiente comando:

```
hisat2-build --seed 123 -p 2 input/Homo_sapiens.GRCh38.dna.chromosome.21.fa resultados/index/Homo_sapiens.GRCh38.dna.chromosome.21
```

A continuación se detalla el significado del comando utilizado, así como sus opciones:

- El comando `'hisat2-build'` es el comando principal de HISAT2 para generar un índice de referencia.
- `'--seed 123'` especifica la semilla de randomización utilizada para la generación de números aleatorios durante la construcción del índice. Se utiliza una semilla para facilitar la reproducibilidad de los datos.
- `'-p 2'` especifica el número de subprocesos que se utilizarán durante la construcción del índice. En este caso, se han utilizado 2 subprocesos.
- `'input/Homo_sapiens.GRCh38.dna.chromosome.21.fa'` es la ubicación del archivo FASTA que contiene la secuencia de ADN del cromosoma 21 del genoma humano, que es el archivo que se utiliza como entrada para construir el índice.
- `'resultados/index/Homo_sapiens.GRCh38.dna.chromosome.21'` es la ubicación y el nombre del archivo de índice resultante que se generará. El índice se almacenará en el directorio "resultados/index" con el nombre "Homo_sapiens.GRCh38.dna.chromosome.21"

Alineamiento

Para el alineamiento de las muestras se utilizaron los siguientes comandos.

Para la muestra SRR479052:

```
hisat2 --new-summary --summary-file resultados/hisat2/SRR479052.hisat2.summary --rna-strandness R --seed 123 --phred33 -p 2 -k 1 -x resultados/index/Homo_sapiens.GRCh38.dna.chromosome.21 -1 input/SRR479052.chr21_1.fastq -2 input/SRR479052.chr21_2.fastq -S resultados/hisat2/SRR479052_chr21.sam
```

Para la muestra SRR479054:

```
hisat2 --new-summary --summary-file resultados/hisat2/SRR479054.hisat2.summary --rna-strandness R --seed 123 --phred33 -p 2 -k 1 -x resultados/index/Homo_sapiens.GRCh38.dna.chromosome.21 -1 input/SRR479054.chr21_1.fastq -2 input/SRR479054.chr21_2.fastq -S resultados/hisat2/SRR479054_chr21.sam
```

A continuación se detalla el significado del comando utilizado, así como también la utilización de las diferentes opciones:

- `'hisat2'` es el comando principal de HISAT2 utilizado para alinear secuencias de lecturas de ARN con un genoma de referencia.
- `'--new-summary'` especifica que se debe generar un archivo de resumen nuevo que contendrá estadísticas sobre el proceso de alineación.
- `'--summary-file resultados/hisat2/SRR479052.hisat2.summary'` especifica la ubicación y el nombre del archivo de resumen que se generará. El archivo se guardará en el directorio "Apartado1/resultados/hisat2" con el nombre "SRR479052.hisat2.summary".
- `'--rna-strandness R'` da información sobre la orientación de la secuencia de ARN. En este caso, se indica que la secuencia es "reverse".
- `'--seed 123'` es una semilla de randomización utilizada para la generación de números aleatorios durante el proceso de alineación que facilita la reproducibilidad de los resultados.

- `'--phred33'` indica que la calidad de las bases está codificada en el formato Phred 33, que es el más común en datos de secuenciación de Ilumina.
- `'-p 2'` indica el número de subprocesos que se utilizan en el proceso de alineación. En este caso se están utilizando dos subprocesos.
- `'-k 1'` especifica el número máximo de alineaciones que se informarán por cada read. En este caso se reportará únicamente la mejor alineación para cada read.
- `'-x resultados/index/Homo_sapiens.GRCh38.dna.chromosome.21'` indica la ubicación y el nombre del archivo de índice que se utilizará en la alineación. En este caso se utilizará el índice previamente construido del cromosoma 21 del genoma humano que se encuentra en la carpeta "resultados/index".
- `'-1 input/SRR479052.chr21_1.fastq'` y `'-2 input/SRR479052.chr21_2.fastq'` son las ubicaciones de los archivos FASTQ que contienen las secuencias de las lecturas pareadas de ARN. Estos archivos contienen las secuencias de la primera y la segunda lectura, respectivamente.
- `'-S resultados/hisat2/SRR479052_chr21.sam'` indica la ubicación y el nombre del archivo SAM que se generará como salida del proceso de alineamiento. El archivo SAM contendrá las alineaciones de las lecturas con el genoma de referencia.

El segundo comando proporcionado tiene las mismas opciones, con la excepción de que se utilizó la muestra SRR479054 en lugar de la muestra SRR479052.

CONTROL DE CALIDAD DEL ALINEAMIENTO

Como se ha mencionado anteriormente, en los archivos .summary generados con los comandos hisat2 se obtienen unas métricas de alineamiento que informan sobre la calidad del alineamiento. Dichas métricas se muestran a continuación.

Muestra SRR479052

HISAT2 summary stats:

Total pairs: 15340

Aligned concordantly or discordantly 0 time: 6663 (43.44%)

Aligned concordantly 1 time: 7061 (46.03%)

Aligned concordantly >1 times: 0 (0.00%)

Aligned discordantly 1 time: 1616 (10.53%)

Total unpaired reads: 13326

Aligned 0 time: 6636 (49.80%)

Aligned 1 time: 6690 (50.20%)

Aligned >1 times: 0 (0.00%)
Overall alignment rate: 78.37%

Según los datos del resumen de HISAT2, se procesaron un total de 15,340 pares de secuencias. De estos, el 43.44% no se alinearon ni concordante ni discordantemente, mientras que el 46.03% se alineó concordantemente una vez. No se observaron alineamientos concordantes múltiples ni discordantes múltiples. Además, se encontró que 1,616 pares se alinearon discordantemente una vez. Además de los pares, se procesaron 13,326 secuencias sin pareja, de las cuales el 49.80% no se alinearon en absoluto, mientras que el 50.20% se alineó una vez. No se registraron alineamientos múltiples para estas secuencias sin pareja. En general, la tasa de alineación global fue del 78.37%.

Muestra SRR479054

HISAT2 summary stats:
Total pairs: 9746
Aligned concordantly or discordantly 0 time: 4043 (41.48%)
Aligned concordantly 1 time: 4852 (49.78%)
Aligned concordantly >1 times: 0 (0.00%)
Aligned discordantly 1 time: 851 (8.73%)
Total unpaired reads: 8086
Aligned 0 time: 4044 (50.01%)
Aligned 1 time: 4042 (49.99%)
Aligned >1 times: 0 (0.00%)
Overall alignment rate: 79.25%

Según el resumen de HISAT2, se procesaron un total de 9,746 pares de secuencias. De estos, el 41.48% no se alinearon ni concordante ni discordantemente, mientras que el 49.78% se alineó concordantemente una vez. No se observaron alineamientos concordantes múltiples ni discordantes múltiples. Además, se encontró que 851 pares se alinearon discordantemente una vez. Además de los pares, se procesaron 8,086 secuencias sin pareja, de las cuales el 50.01% no se alinearon en absoluto, mientras que el 49.99% se alineó una vez. No se registraron alineamientos múltiples para estas secuencias sin pareja. En general, la tasa de alineación global fue del 79.25%.

A continuación se podría hacer un procesado de las muestras con samtools para obtener archivos .bam a partir de los archivos .sam generados, así como ordenar e indexar los archivos

.bam generados. Sin embargo, al tratarse de un paso opcional que no aportará una mayor calidad al resto de los resultados, se ha decidido no realizar este procesado.

CONTEO DE LAS LECTURAS

Para el conteo de las lecturas se utilizó el programa HTSEQ (versión 2.0.5). HTSeq es una herramienta bioinformática utilizada comúnmente para el análisis de datos de secuenciación de ARN (ARN-seq). Su función principal es asignar recuentos de lecturas de secuencias de ARN a diferentes características genómicas, como genes, exones o regiones de interés, a partir de archivos de alineación generados por programas como HISAT2 o TopHat. HTSeq funciona mediante la comparación de las coordenadas de alineación de las lecturas con las anotaciones genómicas proporcionadas en archivos de formato GTF o GFF. Para cada característica genómica, HTSeq determina cuántas lecturas se superponen y asigna este recuento como el número de lecturas alineadas a esa característica. Este recuento se utiliza posteriormente en el análisis diferencial de expresión génica y en la caracterización de la abundancia relativa de transcritos. Además, HTSeq proporciona opciones para manejar lecturas ambiguas y filtrar lecturas basadas en criterios específicos, lo que lo convierte en una herramienta versátil para el análisis de datos de ARN-seq.

A continuación se muestran los comandos utilizados para este paso.

Para la muestra SRR479052:

```
htseq-count --format=sam --stranded=reverse
--mode=intersection-nonempty --minaqual=10 --type=exon
--idattr=gene_id --additional-attr=gene_name
resultados/hisat2/SRR479052_chr21.sam
input/Homo_sapiens.GRCh38.109.chr21.gtf >
resultados/htseq/SRR479052_chr21.htseq
```

Para la muestra SRR479054:

```
htseq-count --format=sam --stranded=reverse
--mode=intersection-nonempty --minaqual=10 --type=exon
--idattr=gene_id --additional-attr=gene_name
resultados/hisat2/SRR479054_chr21.sam
input/Homo_sapiens.GRCh38.109.chr21.gtf >
resultados/htseq/SRR479054_chr21.htseq
```

A continuación se detalla el significado del comando utilizado, así como también la utilización de las diferentes opciones:

- `'htseq-count'` es el comando principal de HTSeq utilizado para el recuento de lecturas.
- `'--format=sam'` especifica el formato de entrada del archivo de alineación como archivo .sam.
- `'--stranded=reverse'` indica que se considera la información de hebra al realizar el recuento y que las lecturas están orientadas en la hebra inversa, de acuerdo con lo indicado en el alineamiento de secuencias.
- `'--mode=intersection-nonempty'` define el modo de recuento. En este caso cuenta las lecturas que intersectan con al menos una parte de una característica genómica (como un exón, por ejemplo).
- `'--minaqual=10'` establece el valor mínimo de calidad de la base de las lecturas para ser tenidas en cuenta en el recuento.
- `'--type=exon'` sirve para indicar el tipo de característica genómica sobre la cual se realizará el recuento. En este caso, se utilizan exones.
- `'--idattr=gene_id'` indica qué atributo del archivo de anotaciones GTF se utilizará para identificar las características genómicas. En este caso, se utiliza el atributo "gene_id".
- `'--additional-attr=gene_name'` permite agregar atributos adicionales al resultado del recuento. En este caso, se añade el nombre del gen.
- 'resultados/hisat2/SRR479052_chr21.sam' indica el archivo de alineación SAM generado en el apartado anterior con HISAT2.
- `'input/Homo_sapiens.GRCh38.109.chr21.gtf'` es la ruta al archivo de anotaciones GTF que contiene la información sobre la estructura genómica.
- `'> resultados/htseq/SRR479052_chr21.htseq'` redirige la salida del comando al archivo denominado "SRR479052_chr21.htseq" en el directorio "resultados/htseq". En ese archivo se almacenarán los resultados del recuento.

El segundo comando proporcionado tiene las mismas opciones, con la excepción de que se utilizó la muestra SRR479054 en lugar de la muestra SRR479052.

Tras ejecutar estos comandos se obtuvieron unas matrices de cuentas que contienen las siguientes columnas:

- El identificador de Ensembl.

- El nombre del gen.
- El número de cuentas que se han asignado.

Para la interpretación de los resultados, se utilizó el programa multiqc (versión 1.19) ejecutando el siguiente comando:

```
multiqc -o resultados/multiqc/ resultados/htseq
```

- `'multiqc'` es el comando principal de multiqc para generar el informe de calidad combinado.
- `'-o resultados/multiqc/'` indica la ubicación donde se guardarán los archivos de salida del informe de calidad.
- `'resultados/htseq'` es la ruta del directorio que contiene los archivos de resultado del conteo de lecturas.

Tras ejecutar este comando se obtuvo un archivo .html en el que se muestra un gráfico (**Figura 5**) en el que se observa la asignación de las cuentas realizadas por HTSEQ.

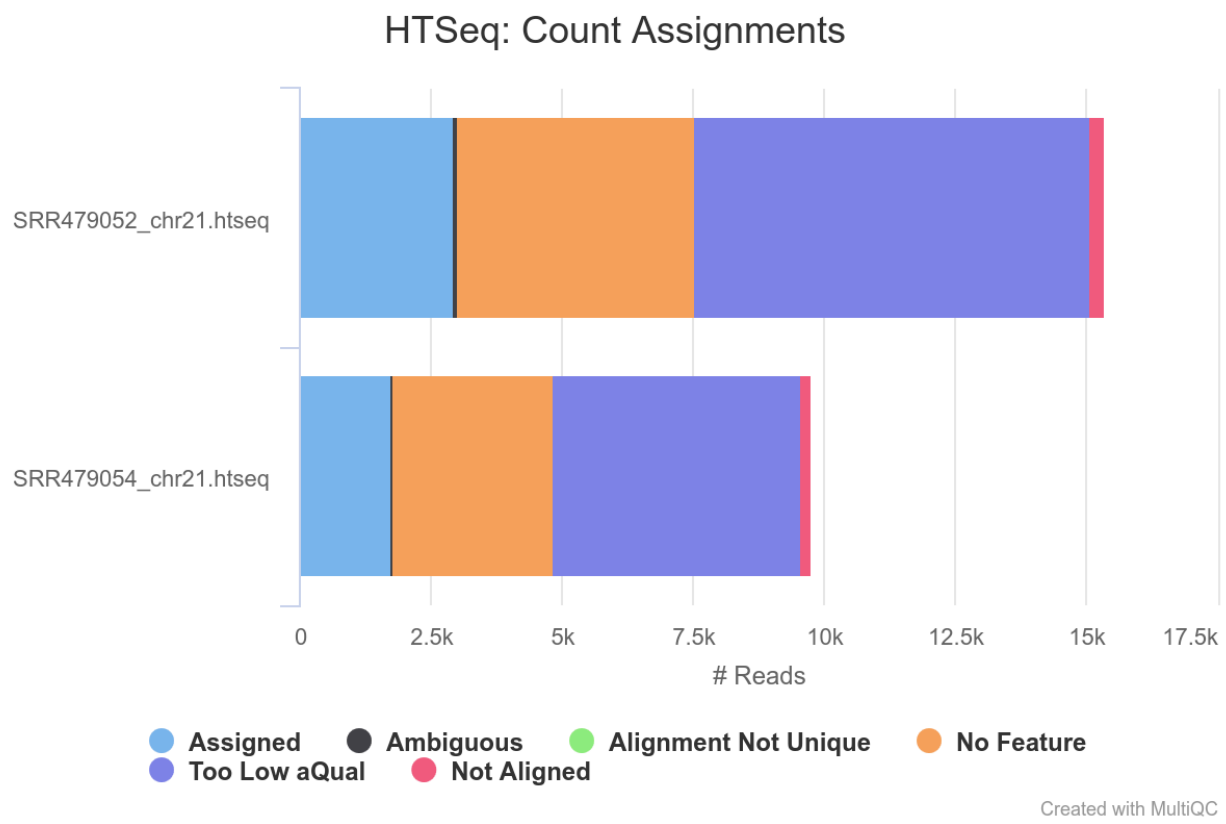

Figura 5. Gráfica obtenida con multiqc con asignación de las cuentas obtenidas por HTSEQ.

Se puede observar que la mayoría de las cuentas no fueron asignadas por tener un score de calidad muy bajo (color morado, “Too Low aQual”) o porque no pudieron ser asignadas a ninguna función (naranja, “No Feature”). En azul claro (“Assigned”) se observan las lecturas que fueron contadas y asignadas a una anotación. Como último comentario, se pueden observar también de forma minoritaria lecturas que no fueron alineadas (rosa, “Not Aligned”) y lecturas ambiguas (negro, “Ambiguous”).

CONCLUSIONES

Como conclusión de este primer apartado, se destaca que se logró alinear aproximadamente el 80% de las secuencias de manera adecuada, aunque muchas no fueron asignadas a anotaciones debido a una baja calidad de las lecturas. Según el análisis de calidad realizado con FastQC, sería recomendable eliminar los adaptadores detectados, así como las secuencias repetidas y los extremos de baja calidad, definidos como aquellos con valores inferiores a 20 en la escala Phred. Esta acción podría mejorar significativamente la calidad de las lecturas y, como resultado, incrementar el número de cuentas asignadas a los genes en las matrices generadas.

APARTADO 2

ANÁLISIS POR DEG

Para averiguar cuáles son los genes diferencialmente expresados entre las muestras pertenecientes al grupo tratado con OHT con respecto al control tras 24h, así como los genes diferencialmente expresados entre las muestras pertenecientes al grupo tratado con DPN con respecto al control tras 24h, se utilizó un análisis DEG. Para ello, se realizó un script en Rstudio denominado DEG.R (en el directorio Apartado2/resultados/DEG) en el que se encuentran todos los comandos utilizados para el desarrollo de este análisis.

Versiones R y paquetes de R

Para este análisis se utilizaron las siguientes versiones de R y sus paquetes:

- Versión R: 4.3.3
- Versión DESeq2: 1.42.1
- Versión tidyverse: 2.0.0
- Versión pheatmap: 1.0.12
- Versión RColorBrewer: 1.1.3
- Versión vsn: 3.70.0

Preprocesado de los datos

Primero se cargaron los archivos “rawcounts.tsv” y “metadata.tsv” proporcionados en el directorio “input”. Además, se trataron los datos de forma que el nombre de las filas del archivo “metadata.tsv” coincidieran con los nombres de las columnas de la matriz de cuentas “rawcounts.tsv”. Por último, se generó una nueva variable denominada “group” que combina las variables “agent” y “time”.

Creación de DESeqDataSet

A continuación se generó el objeto DESeqDataSet, guardado en una variable denominada dds, utilizando como diseño la modelización por la variable “patient” y la variable “group” creada anteriormente. La creación de esta variable permite tratar el tiempo y el tratamiento, sin necesidad de tener que introducir en el diseño la fórmula de interacción entre las dos variables.

```
dds <- DESeqDataSetFromMatrix(countData = raw_data,
                              colData = experiment_data,
```

```
                                design = ~ patient + group)
```

A continuación, se eliminaron de este dataset los genes que tuvieran un número de lecturas inferior a 10, y se generó un nuevo dataset (dds2) en el que estos genes con muy pocas lecturas no aparecen. De esta forma, el número de genes a tratar se redujo de 53160 genes a 24416 genes.

```
keep <- rowSums(counts(dds)) >= 10
dds2 <- dds[keep, ]
```

Análisis exploratorio

Antes de realizar el análisis de expresión diferencial es necesario revisar los datos para asegurarse de que el posterior análisis obtenga los mejores resultados.

Transformación estabilizadora de la varianza (VST)

La función vst() (variance stabilizing transformation) es una función utilizada en el paquete DESeq2 de R para realizar una transformación de estabilización de varianza en datos de expresión génica. Esta transformación es útil para reducir la heterocedasticidad de los datos, es decir, para hacer que la varianza de los datos sea más constante en diferentes niveles de expresión génica. Esto es importante porque muchos métodos estadísticos y modelos asumen que la varianza de los datos es constante, lo que puede no ser cierto en el caso de datos de expresión génica.

Se realizó la transformación estabilizadora de la varianza en los datos de expresión contenidos en el objeto "dds2" y se almacenó esta transformación en una nueva variable denominada "vsd". La opción `'blind = TRUE'` indica que la transformación se realizará sin conocimiento previo de las clases de muestra.

```
vsd <- vst(dds2, blind = TRUE)
```

Para comprobar el efecto de esta transformación, se compararon los efectos de la transformación VST frente a los datos de expresión normalizados. Para ello, primero se normalizaron los datos:

```
normal_data <- normTransform(dds2)
```

Y a continuación se generaron gráficas de dispersión de media frente a desviación estándar:

```
normal_graph <- meanSdPlot(assay(normal_data))
```

```
vsd_graph <- meanSdPlot(assay(vsd))
```

Con lo que se obtuvieron las siguientes gráficas:

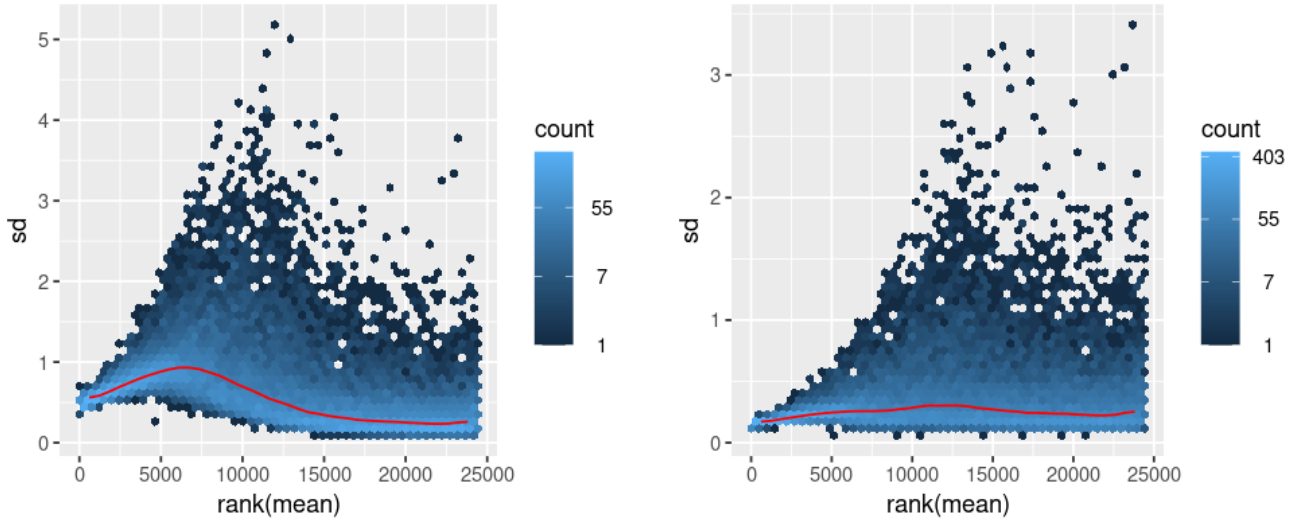

Figura 6. Comparación de los efectos causados por la transformación de la varianza. A la izquierda se encuentra la gráfica correspondiente a los datos normalizados sin transformación de la varianza, y a la derecha se encuentra la gráfica correspondiente a los datos tras realizarse la transformación de la varianza.

En la **Figura 6** se aprecia que los datos se estabilizan, ya que la línea roja correspondiente a la relación entre la media y la varianza se aplana en la gráfica de la derecha (correspondiente a los datos tras la transformación de la varianza).

Análisis de componentes principales (PCA)

Para visualizar la estructura de los datos de expresión génica e identificar patrones biológicos se realizó un PCA con las muestras preprocesadas (a partir del objeto "vsd").

```
plotPCA(vsd, intgroup = "patient")
plotPCA(vsd, intgroup = "group")
plotPCA(vsd, intgroup = "agent")
plotPCA(vsd, intgroup = "time")
```

Obteniéndose las imágenes mostradas en la **Figura 7**.

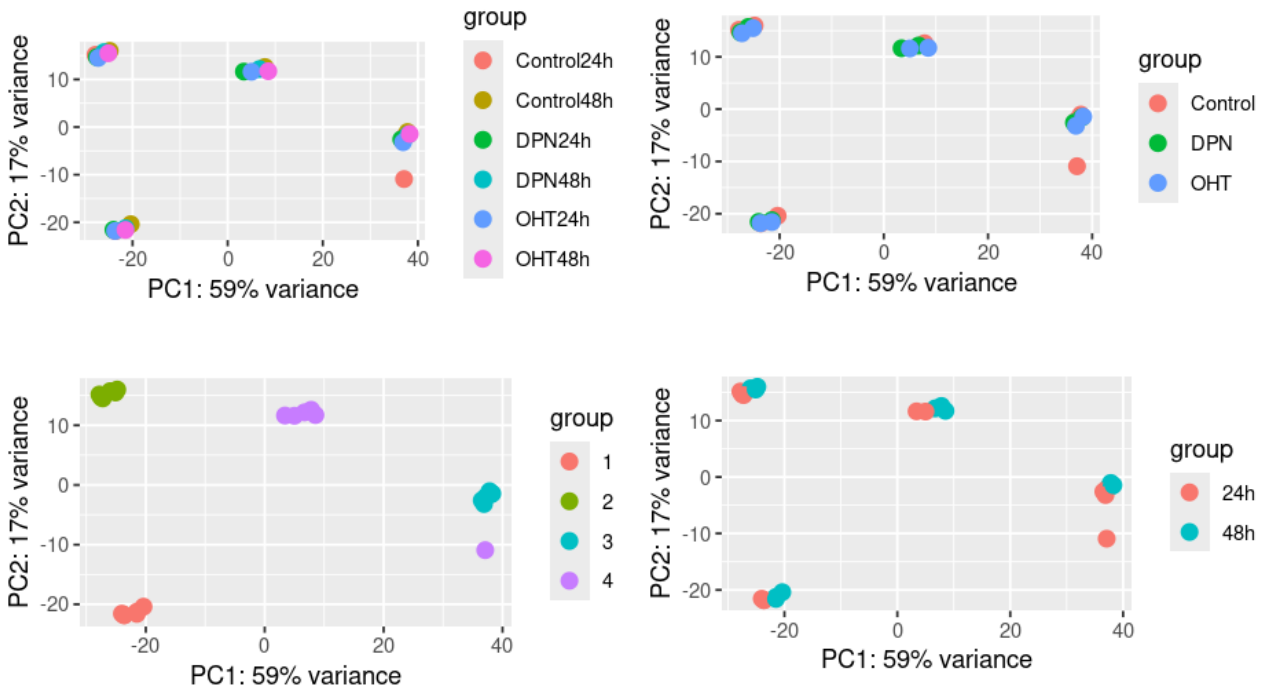

Figura 7. PCA de las muestras. Arriba a la izquierda es la muestra correspondiente con el grupo "group", arriba a la derecha la muestra correspondiente con el grupo "agent", abajo a la izquierda la muestra correspondiente con el grupo "patient" y abajo a la derecha la muestra correspondiente con el grupo "time".

Se puede observar cómo en las cuatro imágenes se observa un outlier que corresponde con el paciente 4, que tiene tratamiento control y tiempo 24h (pertenece al grupo "Control24h").

Matriz de distancias

Similarmente, se pueden comprobar los resultados del PCA realizando una matriz de distancias. Una matriz de distancias es una representación matricial que describe las distancias o similitudes entre pares de elementos en un conjunto de datos.

```
sampleDists <- dist(t(assay(vsd)))
sampleDistMatrix <- as.matrix( sampleDists )
rownames(sampleDistMatrix) <- paste( vsd$patient, vsd$group, sep
= " - " )
colnames(sampleDistMatrix) <- NULL
colors <- colorRampPalette( rev(brewer.pal(9, "Blues")) )(255)
pheatmap(sampleDistMatrix,
         clustering_distance_rows = sampleDists,
         clustering_distance_cols = sampleDists,
         col = colors)
```

De esta forma, se obtuvo la **Figura 8**.

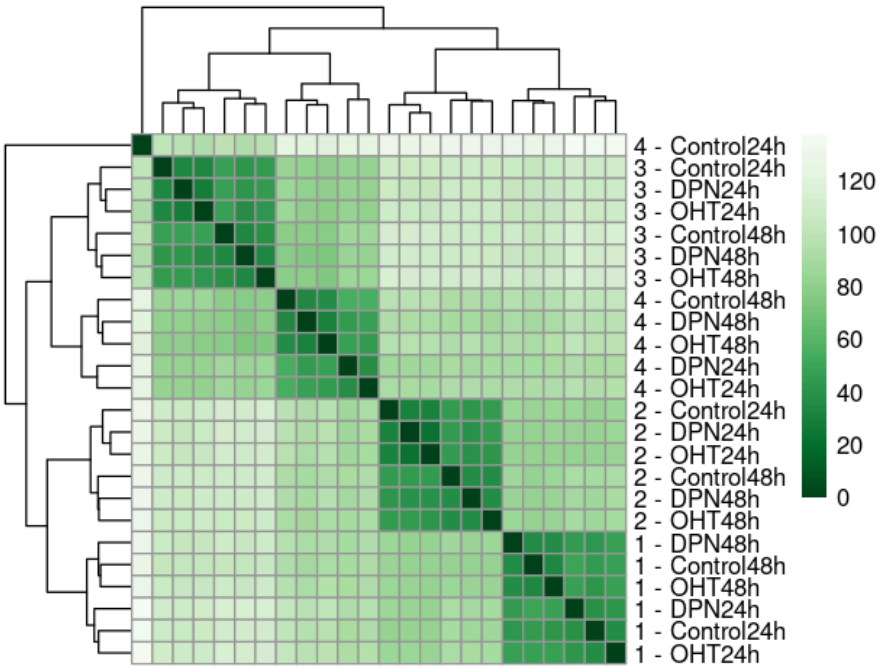

Figura 8. Mapa de calor de la matriz de distancias.

Esta matriz nos permite observar que los pacientes se agrupan de manera coherente entre sí. Además, dentro de cada paciente, los diferentes tiempos de muestreo tienden a agruparse entre sí (por ejemplo, 24 horas versus 48 horas). Respecto a las relaciones entre pacientes, parece que los pacientes 1 y 2 están más relacionados entre sí, al igual que los pacientes 3 y 4. Además, al analizar el posible valor atípico, confirmamos su presencia utilizando este método y observamos que, al igual que en el PCA, se encuentra más próximo a los datos del paciente 3 que a los de su propio grupo.

Análisis de expresión diferencial

Antes de realizar el análisis de expresión diferencial se tomó la decisión de eliminar el outlier, ya que se trata de un valor atípico que puede afectar a la calidad de los resultados y hacer que el análisis sea menos representativo.

Para ello, se podrían eliminar todos los datos relativos al paciente 4, o bien podría eliminarse únicamente el outlier. En este caso se utilizó esta segunda aproximación, ya que a pesar de no ser la técnica más idónea permite evaluar la influencia que tiene ese único dato en el resto de datos del análisis. En caso de eliminar el grupo entero, no se podría hacer esta evaluación.

```
patient_outlier <- which(experiment_data$patient == 4)
agent_outlier <- which(experiment_data$agent == "Control")
time_outlier <- which(experiment_data$time == "24h")
patient_agent <- intersect(patient_outlier, agent_outlier)
patient_time <- intersect(patient_outlier, time_outlier)
outlier <- intersect(patient_agent, patient_time)
experiment_data_clean <- experiment_data[-outlier,]
raw_data_clean <- raw_data[ ,-outlier]
```

Una vez habiendo eliminado el outlier, se generó un nuevo DESeqDataSet con las matrices de cuentas y metadatos nuevos sin el outlier.. Además, se repitió el preprocesado de los datos llevado a cabo anteriormente, eliminando los genes cuyas lecturas fueran menores a 10. Así, de los 53160 genes iniciales, se obtuvieron 23233 genes.

```
dds_outlier <- DESeqDataSetFromMatrix(countData = raw_data_clean,
                                      colData                          =
experiment_data_clean,
                                      design = ~ patient + group)
dds_outlier
keep <- rowSums(counts(dds_outlier)) >= 10
dds2_outlier <- dds_outlier[keep, ]
```

De nuevo, es necesario realizar la transformación estabilizadora de la varianza. Se realizó la transformación estabilizadora de la varianza en los datos de expresión contenidos en el objeto "dds2_outlier" y se almacenó esta transformación en una nueva variable denominada "vsd_outlier". La opción '`blind = TRUE`' indica que la transformación se realizará sin conocimiento previo de las clases de muestra.

```
vsd_outlier <- vst(dds2_outlier, blind = TRUE)
```

Para comprobar el efecto de esta transformación, se compararon los efectos de la transformación VST frente a los datos de expresión normalizados. Para ello, primero se normalizaron los datos:

```
normal_data_outlier <- normTransform(dds2_outlier)
```

Y a continuación se generaron gráficas de dispersión de media frente a desviación estándar:

```
normal_graph_outlier <- meanSdPlot(assay(normal_data_outlier))
vsd_graph_outlier <- meanSdPlot(assay(vsd_outlier))
```

Con lo que se obtuvieron las siguientes gráficas:

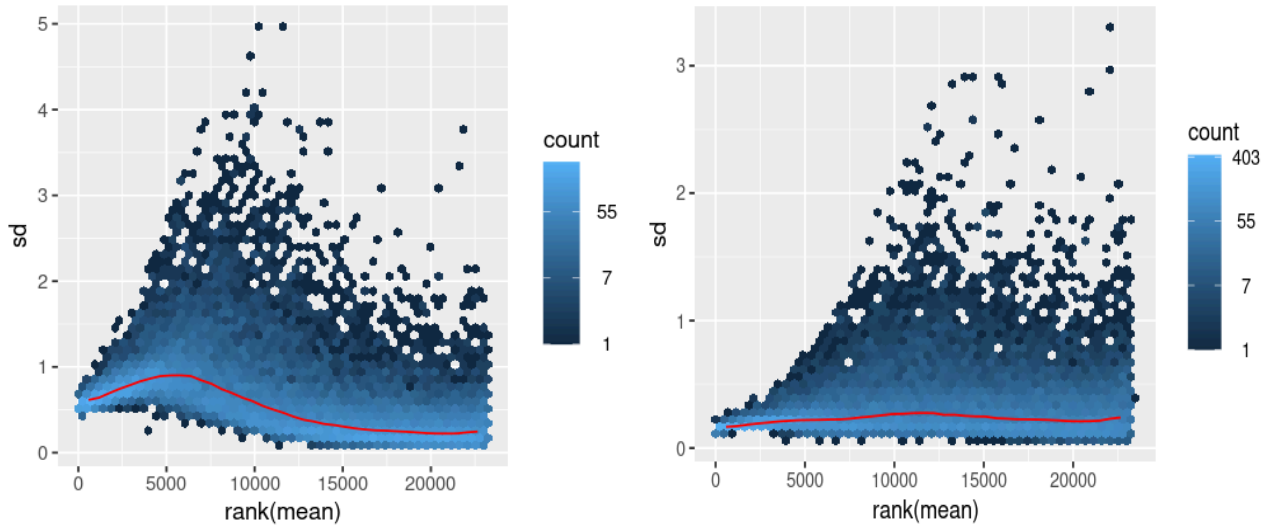

Figura 9. Comparación de los efectos causados por la transformación de la varianza. A la izquierda se encuentra la gráfica correspondiente a los datos normalizados sin transformación de la varianza, y a la derecha se encuentra la gráfica correspondiente a los datos tras realizarse la transformación de la varianza.

Se puede observar en la **Figura 9** que los datos se estabilizan, ya que la línea roja correspondiente a la relación entre la media y la varianza se aplana en la gráfica de la derecha (correspondiente a los datos tras la transformación de la varianza).

A continuación es necesario volver a realizar el PCA y la matriz de distancias, para comprobar que el outlier ha sido eliminado correctamente.

```
plotPCA(vsd_outlier, intgroup = "patient")
plotPCA(vsd_outlier, intgroup = "group")
plotPCA(vsd_outlier, intgroup = "agent")
plotPCA(vsd_outlier, intgroup = "time")
```

Obteniéndose las imágenes mostradas en la **Figura 10**.

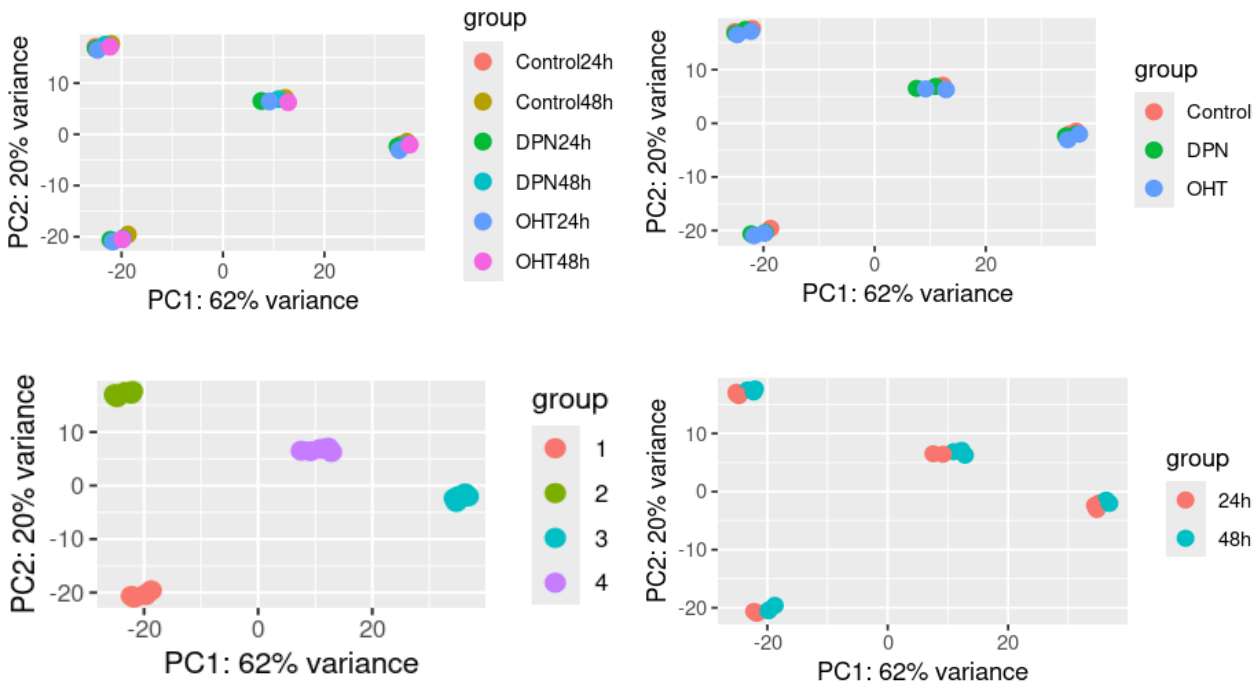

Figura 10. PCA de las muestras sin el outlier. Arriba a la izquierda es la muestra correspondiente con el grupo "group", arriba a la derecha la muestra correspondiente con el grupo "agent", abajo a la izquierda la muestra correspondiente con el grupo "patient" y abajo a la derecha la muestra correspondiente con el grupo "time".

Se puede observar que la separación entre los diferentes grupos se sigue manteniendo pero no existe ningún outlier. A continuación se realizó la nueva matriz de distancias para comprobar por última vez que la relación entre los diferentes grupos no se ha visto afectada por la eliminación del outlier.

```
sampleDists <- dist(t(assay(vsd)))
sampleDistMatrix <- as.matrix( sampleDists )
rownames(sampleDistMatrix) <- paste( vsd$patient, vsd$group, sep
= " - " )
colnames(sampleDistMatrix) <- NULL
colors <- colorRampPalette( rev(brewer.pal(9, "Blues")) )(255)
pheatmap(sampleDistMatrix,
         clustering_distance_rows = sampleDists,
         clustering_distance_cols = sampleDists,
         col = colors)
```

De esta forma, se obtiene la **Figura 11**.

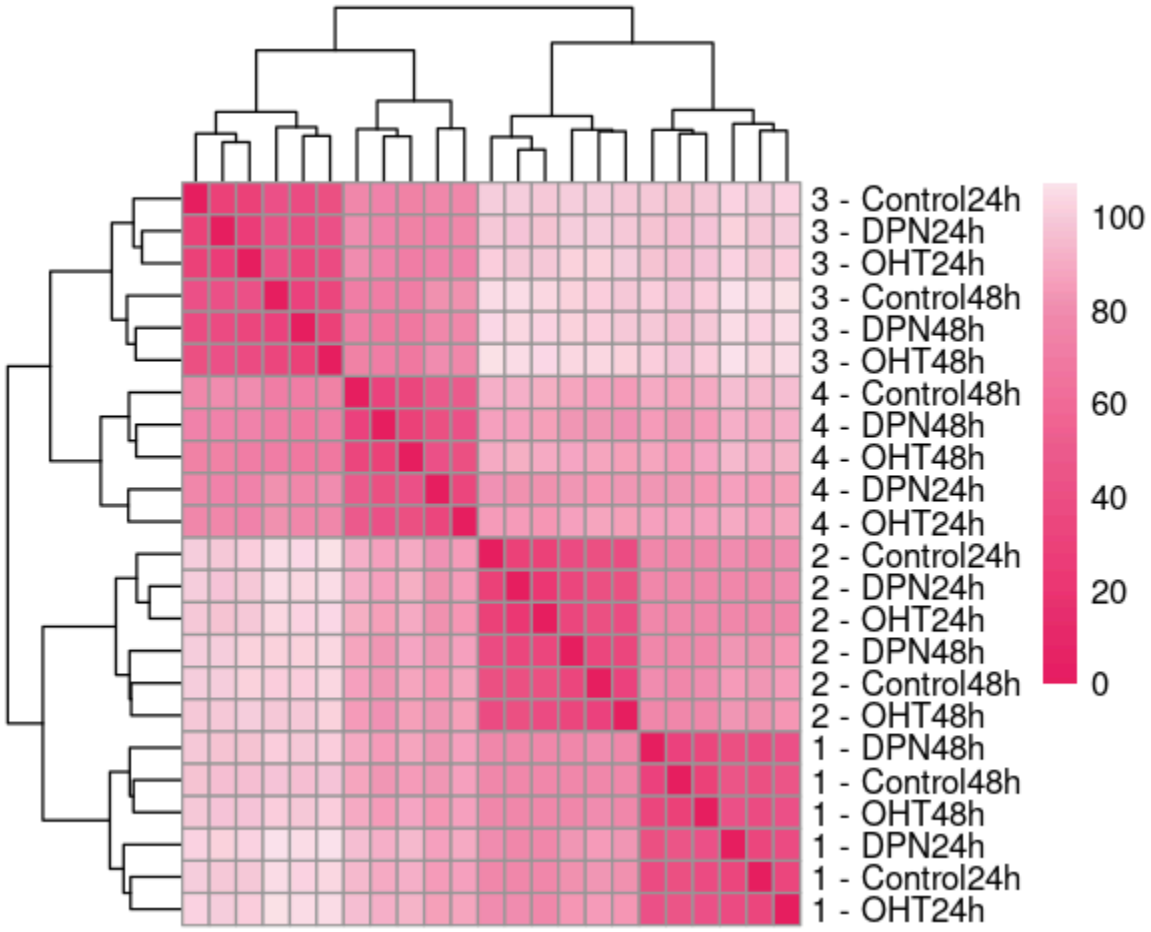

Figura 11. Mapa de calor de la matriz de distancias.

Esta matriz nos permite observar que los pacientes se agrupan de manera coherente entre sí. Además, dentro de cada paciente, los diferentes tiempos de muestreo tienden a agruparse entre sí (por ejemplo, 24 horas versus 48 horas). Respecto a las relaciones entre pacientes, parece que los pacientes 1 y 2 están más relacionados entre sí, al igual que los pacientes 3 y 4. Además, en este caso el outlier ha sido eliminado, por lo que la relación entre los pacientes es coherente con el tratamiento que están recibiendo.

El siguiente paso es generar un nuevo objeto DESeqDataSet sin el outlier y comprobar gráficamente la distribución de la estimación de la dispersión y los resultados de la expresión diferencial en el MA-plot. Los resultados obtenidos se observan en la **Figura 12**.

```
dds3_outlier <- DESeq(dds2_outlier, test = "Wald")
plotDispEsts(dds3_outlier)
plotMA(dds3_outlier)
```

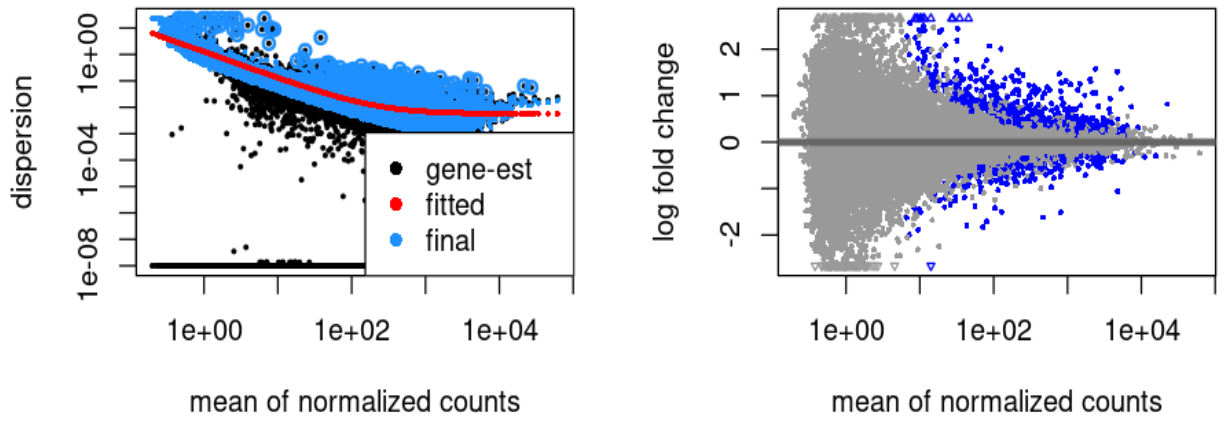

Figura 12. Estimación de la dispersión y resultados de expresión diferencial. La figura de la izquierda muestra la estimación de la dispersión, mientras que la figura de la derecha muestra la expresión diferencial mediante un MA-plot. Los valores coloreados del MA-plot corresponden con genes diferencialmente expresados.

Se puede observar en el MA-plot que apenas hay genes con log fold change negativo.

A continuación, se analizó el efecto que tiene el tratamiento con DPN frente al control en 24h sin aplicar un umbral.

```
res_DPN_outlier <- results(object = dds3_outlier,
                           contrast = c("group", "Control24h", "DPN24h"),
                           alpha = 0.05,
                           pAdjustMethod = "BH"
                           )
summary(res_DPN_outlier)
```

Con lo que se obtuvieron los siguientes resultados:

```
out of 23233 with nonzero total read count
adjusted p-value < 0.05
LFC > 0 (up)       : 3, 0.013%
LFC < 0 (down)     : 2, 0.0086%
outliers [1]       : 0, 0%
low counts [2]     : 0, 0%
(mean count < 0)
[1] see 'cooksCutoff' argument of ?results
[2] see 'independentFiltering' argument of ?results
```

Se puede observar que hay 5 genes diferencialmente expresados.

Se realizó el mismo análisis, con la excepción de que en este caso sí se aplicó un umbral (LFC=1).

```
res_DPN_lfc1_outlier <- results(object = dds3_outlier,
                        contrast = c("group", "Control24h", "DPN24h"),
                        alpha = 0.05,
                        lfcThreshold = 1,
                        pAdjustMethod = "BH"
)
summary(res_DPN_lfc1_outlier)
```

Con lo que se obtuvieron los siguientes resultados:

```
out of 23233 with nonzero total read count
adjusted p-value < 0.05
LFC > 1.00 (up)     : 0, 0%
LFC < -1.00 (down) : 0, 0%
outliers [1]        : 0, 0%
low counts [2]      : 0, 0%
(mean count < 0)
[1] see 'cooksCutoff' argument of ?results
[2] see 'independentFiltering' argument of ?results
```

En este caso, no hay ningún gen diferencialmente expresado.

Se repitió este mismo proceso para el tratamiento con OHT para las muestras sin aplicar umbral:

```
res_OHT_outlier <- results(object = dds3_outlier,
                        contrast = c("group", "Control24h", "OHT24h"),
                        alpha = 0.05,
                        pAdjustMethod = "BH"
)
summary(res_OHT_outlier)
```

Obteniendo los siguientes resultados:

```
out of 23233 with nonzero total read count
adjusted p-value < 0.05
LFC > 0 (up)        : 0, 0%
LFC < 0 (down)      : 0, 0%
outliers [1]        : 0, 0%
low counts [2]      : 0, 0%
```

```
(mean count < 0)
[1] see 'cooksCutoff' argument of ?results
[2] see 'independentFiltering' argument of ?results
```

Se realizó el mismo análisis, con la excepción de que en este caso sí se aplicó un umbral (LFC=1).

```
res_OHT_lfc1_outlier <- results(object = dds3_outlier,
                                contrast = c("group", "Control24h", "OHT24h"),
                                alpha = 0.05,
                                lfcThreshold = 1,
                                pAdjustMethod = "BH"
)
summary(res_OHT_lfc1_outlier)
```

Obteniendo los siguientes resultados:

```
out of 23233 with nonzero total read count
adjusted p-value < 0.05
LFC > 1.00 (up)    : 0, 0%
LFC < -1.00 (down) : 0, 0%
outliers [1]       : 0, 0%
low counts [2]     : 0, 0%
(mean count < 0)
[1] see 'cooksCutoff' argument of ?results
[2] see 'independentFiltering' argument of ?results
```

En este caso, no se obtuvo ningún gen diferencialmente expresado, tanto en el caso en el que se aplica el filtro LFC como en el caso en el que no se aplica dicho filtro.

Visualización de resultados

Estos resultados se pueden observar gráficamente mediante la realización de un “heatmap”, que se puede observar en la **Figura 13**.

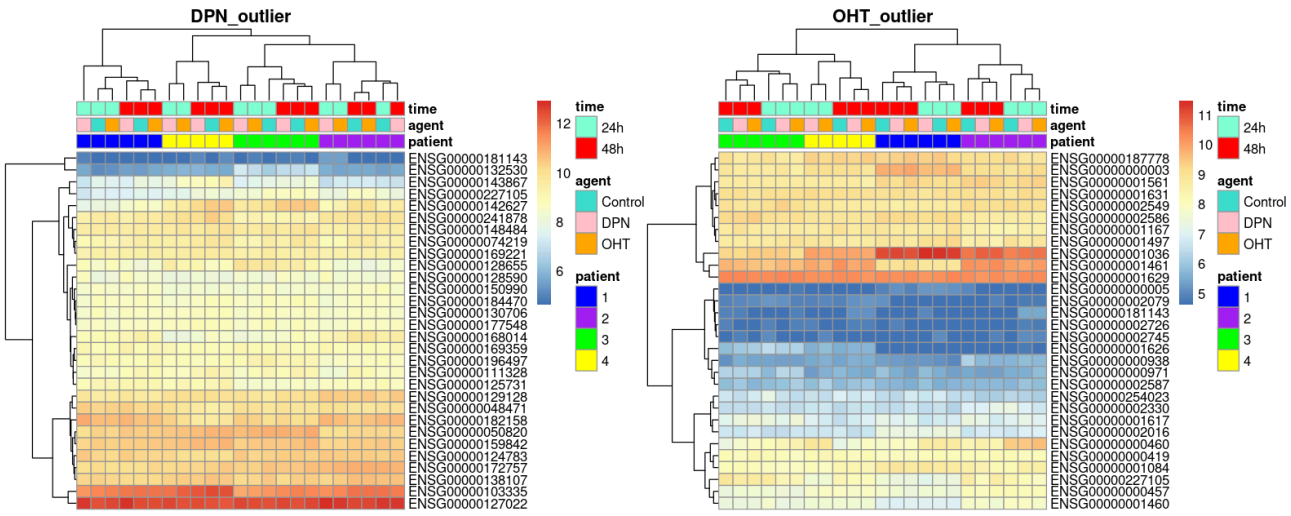

Figura 13. Heatmap de todos los genes diferencialmente expresados.

Se observa que no hay ningún grupo de genes que se encuentren diferencialmente expresados.

ANÁLISIS CON GSEA

Versiones R y paquetes de R

Para este análisis se han utilizado las siguientes versiones de R y sus paquetes:

- Versión R: 4.3.3
- Versión DESeq2: 1.42.1
- Versión tidyverse: 2.0.0
- Versión VennDiagram: 1.7.3

Creación del archivo .rnk

Para hacer el análisis con GSEA pre-ranked, son necesarios dos archivos:

1. Archivo .gmt que contiene los datos de expresión génica para las muestras.
2. Archivo .rnk. Es un archivo de texto que contiene una lista de genes ordenados y la puntuación de clasificación correspondiente. En este trabajo se utilizará el logFold Change (LFC), ya que es el parámetro que se ha utilizado en el análisis DEG.

Para la generación de este archivo .rnk se cargó el objeto dds3.rds generado en el apartado anterior (antes de eliminar el outlier). Después, se realizó un contraste entre DPN24h y Control24h.

```
dds <- readRDS("input/dds3.rds")
res <- results(dds, alpha = 0.05, contrast = c("group", "DPN24h", "Control24h"))
summary(res)
```

Con lo que se obtuvo el siguiente resumen:

```
out of 24416 with nonzero total read count
adjusted p-value < 0.05
LFC > 0 (up)       : 13, 0.053%
LFC < 0 (down)     : 74, 0.3%
outliers [1]       : 0, 0%
low counts [2]     : 10888, 45%
(mean count < 28)
[1] see 'cooksCutoff' argument of ?results
[2] see 'independentFiltering' argument of ?results
```

Para mejorar la precisión de las estimaciones de los parámetros es recomendable reducir los datos. Para ello, se utilizó la función `lfcShrink` con los parámetros que se muestran a continuación:

```
res.ape <- lfcShrink(dds = dds, coef = "group_DPN24h_vs_Control24h", type = "apeglm", res = res)
```

- `'dds'` especifica el objeto DESeqDataSet que contiene los datos de expresión génica y la información del diseño experimental. En este caso, el objeto es dds_GSEA.
- `'coef'` especifica qué coeficiente del diseño se utilizará para calcular los valores de log-fold change (LFC).
- `'type'` sirve para especificar el método que se utilizará para realizar el ajuste de los LFC. En este caso, se utiliza "apelgm". Este método aplica una estimación Bayesiana del LFC, que es robusta y puede proporcionar estimaciones más precisas.
- `'res'` indica los resultados del análisis diferencial previo.

Se puede observar de forma gráfica la diferencia entre los datos reducidos y sin reducir en la **Figura 14**.

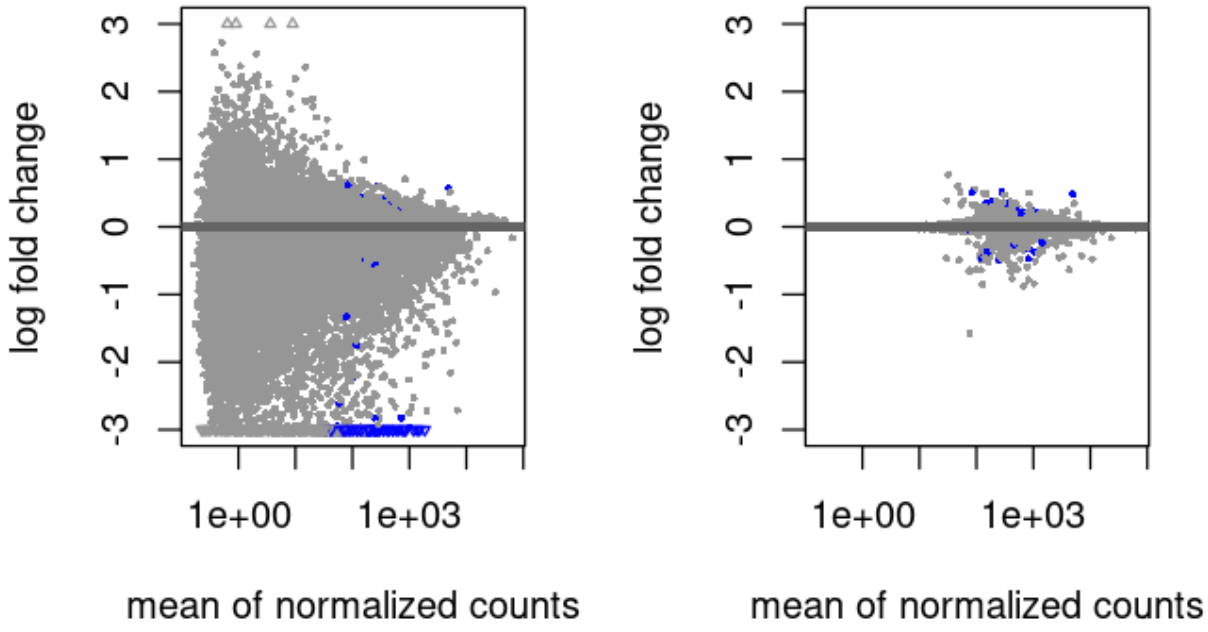

Figura 14. Comparación de los MAplot de los datos reducidos frente a los datos sin reducir. Los datos reducidos se encuentran a la izquierda, mientras que los datos sin reducir se encuentran a la derecha.

Por último, se guardaron estos datos en un archivo con extensión .rnk.

```
rnk <- data.frame(Feature = rownames(res.ape), LFC = res.ape$log2FoldChange)
write.table(rnk, file = "input/DPN_ranked.rnk", sep = "\t", quote = FALSE,col.names = FALSE, row.names = FALSE)
```

Análisis GSEA

Para el análisis GSEA se utilizó la aplicación para escritorio GSEA, y se cargaron en la misma los dos archivos necesarios. Después, se procesó la información con la opción pre-ranked. Para ello, se ejecutó el siguiente comando en la consola de bash, teniendo en cuenta que el directorio de trabajo es "GSEA_Linux_4.3.3"

```
bash gsea-cli.sh GSEAPreranked -gmx ftp.broadinstitute.org://pub/gsea/msigdb/human/gene_sets/h.all.v2023.2.Hs.symbols.gmt,/home/vant/Documentos/Master/Transcriptomi
```

```
ca/ejercicio_final/Apartado2/input/DPN_response.gmt -collapse
Collapse -mode Abs_max_of_probes -norm meandiv -nperm 1000
-rnd_seed 149 -rnk
/home/vant/Documentos/Master/Transcriptomica/ejercicio_final/Apa
rtado2/input/DPN-Control_24h.rnk -scoring_scheme weighted
-rpt_label my_analysis -chip
ftp.broadinstitute.org://pub/gsea/msigdb/human/annotations/Human
_Ensembl_Gene_ID_MSigDB.v2023.2.Hs.chip -create_svgs false
-include_only_symbols true -make_sets true -plot_top_x 20
-set_max 500 -set_min 15 -zip_report false -out
/home/vant/Documentos/Master/Transcriptomica/ejercicio_final/Apa
rtado2/resultados/GSEA/report
```

Este comando ejecuta una análisis de GSEA (Gene Set Enrichment Analysis) prerankeado utilizando el script `gsea-cli.sh`. Aquí está una explicación de los diferentes parámetros utilizados en el comando:

- `'GSEAPreranked'` especifica que se está realizando un análisis de GSEA prerankeado.
- `'-gmx'` especifica los archivos GMT que contienen los conjuntos de genes. En este caso, se proporcionan dos archivos GMT separados por coma. Uno se descarga desde `ftp.broadinstitute.org` y el otro se encuentra en la ruta `/home/vant/Documentos/Master/Transcriptomica/ejercicio_final/Apartado2/input/DPN_response.gmt`.
- `'-collapse'` especifica cómo se colapsarán los genes múltiples que están representados por una sola etiqueta. En este caso, se utiliza "Collapse".
- `'-mode'` especifica el modo de manejo de múltiples sondas. En este caso, se utiliza "Abs_max_of_probes".
- `'-norm'` especifica el método de normalización de datos. Aquí, se utiliza "meandiv".
- `'-nperm'` especifica el número de permutaciones a realizar para calcular los valores de p. En este caso, se utilizan 1000 permutaciones.
- `'-rnd_seed'` especifica la semilla aleatoria para la reproducibilidad de los resultados. Aquí, se utiliza 149 como semilla.
- `'-rnk'` especifica el archivo de clasificación predefinido (en formato .rnk). Aquí, se proporciona la ruta del archivo `/home/vant/Documentos/Master/Transcriptomica/ejercicio_final/Apartado2/input/DPN-Control_24h.rnk`.

- ‘-scoring_scheme’ especifica el esquema de puntuación. Aquí, se utiliza "weighted".
- ‘-rpt_label’ especifica una etiqueta para el informe de salida. Aquí, se utiliza "my_analysis".
- ‘-chip’ especifica el archivo de chip utilizado para mapear los identificadores de genes a símbolos de genes. Aquí, se descarga desde `ftp.broadinstitute.org`.
- ‘-create_svgs’ especifica si se deben crear archivos SVG para las visualizaciones. Aquí, se establece en "false".
- ‘-include_only_symbols’ especifica si se deben incluir sólo símbolos de genes en el análisis. Aquí, se establece en "true".
- ‘-make_sets’ especifica si se deben generar conjuntos de genes. Aquí, se establece en "true".
- ‘-plot_top_x’ especifica cuántos genes principales se deben incluir en los gráficos de enriquecimiento. Aquí, se establece en 20.
- ‘-set_max’ especifica el número máximo de genes permitidos en un conjunto de genes. Aquí, se establece en 500.
- ‘-set_min’ especifica el número mínimo de genes permitidos en un conjunto de genes. Aquí, se establece en 15.
- ‘-zip_report’ especifica si se debe comprimir el informe de salida en formato zip. Aquí, se establece en "false".
- ‘-out’ especifica la ubicación del directorio de salida para el informe de GSEA. Aquí, se establece en `/home/vant/Documentos/Master/Transcriptomica/ejercicio_final/Apartado2/resultados/GSEA/report`.

Resultados

En el archivo “index.html” generado en el directorio “my_analysis.GseaPreranked.1711477851437” se generó un índice con los resultados obtenidos del análisis GSEA. El informe proporciona detalles sobre el enriquecimiento de conjuntos de genes en dos fenotipos distintos, uno identificado como "perturbed" (na_pos) y otro como "unperturbed" (na_neg). Para el fenotipo "perturbed", se identificaron 19 de los 50 conjuntos de genes como regulados al alza, de los cuales 6 resultaron significativamente enriquecidos con un FDR (tasa de falso positivo) menor al 25%. Además, 1 conjunto de genes mostró un p-valor nominal menor al 1%, y 4 conjuntos mostraron un p-valor nominal menor al 5%. Se proporcionan detalles adicionales sobre los resultados de enriquecimiento en diferentes formatos, así como una guía para interpretar los resultados. Por otro lado, para el fenotipo

"unperturbed", se identificaron 31 de los 50 conjuntos de genes como regulados al alza, de los cuales 19 resultaron significativamente enriquecidos con un FDR menor al 25%. Además, 9 conjuntos de genes mostraron un p-valor nominal menor al 1%, y 17 conjuntos mostraron un p-valor nominal menor al 5%. Se proporcionan detalles adicionales sobre los resultados de enriquecimiento en diferentes formatos, así como información sobre el conjunto de genes utilizado en el análisis, marcadores génicos y estadísticas globales y gráficos. También se incluyen los parámetros utilizados para el análisis.

DPN perturbed

Para acceder a la información de este fenotipo, se utilizó el archivo "gsea_report_for_na_pos_1711477851437.tsv" presente en el directorio "Apartado2/resultados/GSEA/my_analysis.GseaPreranked.1711477851437". Como se puede observar en la **Tabla 3**, se identificaron 19 gene sets regulados al alza, de los cuales 6 resultaron significativamente enriquecidos (FDR < 0.25). Sin embargo, el FDR debería ser más estricto debido a que los gene sets han sido permutados. De esta forma, si se considera que el FDR ha de ser menor que 0.05 para que pueda considerarse que los gene sets han sido significativamente enriquecidos, no hay ningún gene set en este fenotipo que cumpla con esta condición.

Tabla 3. Información del análisis de GSEA para los gene sets perturbed.

Name	NOM.p.val	FDR.q.val
HALLMARK_OXIDATIVE_PHOSPHORYLATION	0.001443001	0.07745939
HALLMARK_PROTEIN_SECRETION	0.025196850	0.20875295
HALLMARK_INTERFERON_ALPHA_RESPONSE	0.036529680	0.19217512
HALLMARK_HEME_METABOLISM	0.022091310	0.14413133
HALLMARK_UNFOLDED_PROTEIN_RESPONSE	0.051873200	0.17718957
HALLMARK_MTORC1_SIGNALING	0.054363377	0.24426074
HALLMARK_BILE_ACID_METABOLISM	0.151424290	0.40300250
HALLMARK_GLYCOLYSIS	0.112408760	0.35482258
HALLMARK_PEROXISOME	0.244343890	0.47212720

HALLMARK_INTERFERON_GAMMA_RESPONSE	0.247851000	0.48109046
HALLMARK_ADIPOGENESIS	0.397608370	0.69960564
HALLMARK_MYC_TARGETS_V1	0.489082960	0.77263020
HALLMARK_PI3K_AKT_MTOR_SIGNALING	0.497822940	0.73948130
HALLMARK_ANDROGEN_RESPONSE	0.516516500	0.71967860
HALLMARK_PANCREAS_BETA_CELLS	0.534426200	0.69133866
HALLMARK_E2F_TARGETS	0.792134800	0.89037630
HALLMARK_G2M_CHECKPOINT	0.813559300	0.87495810
HALLMARK_WNT_BETA_CATENIN_SIGNALING	0.895052500	1.00000000
HALLMARK_IL6_JAK_STAT3_SIGNALING	0.971768200	0.97187835

En el archivo “gsea_report_for_na_pos_1711477851437.html” presente en el mismo directorio se puede observar la tabla completa, así como acceder al apartado de “Detalles” de los gene sets para obtener más información.

Debido al propósito de este informe, se va a analizar el gene set HALLMARK_OXIDATIVE_PHOSPHORYLATION, para mostrar las gráficas características y sus interpretaciones, ya que todos los gene sets de la **Tabla 3** tienen gráficas similares.

Gene set HALLMARK_OXIDATIVE_PHOSPHORYLATION

Los datos mostrados a continuación se han obtenido del archivo “HALLMARK_OXIDATIVE_PHOSPHORYLATION.html” presente en el mismo directorio. A continuación (**Figura 15**) se muestran los gráficos característicos del análisis GSEA para el gene set HALLMARK_OXIDATIVE_PHOSPHORYLATION.

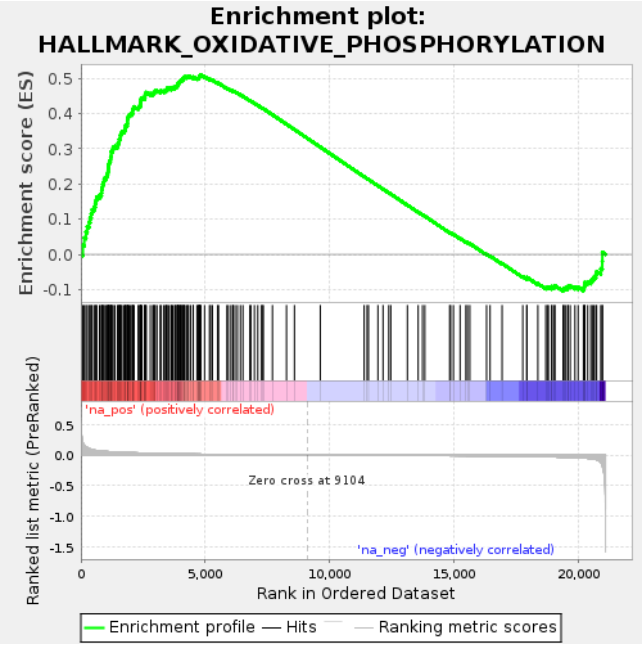

Figura 15. Gráfica de enriquecimiento para el gene set HALLMARK_OXIDATIVE_PHOSPHORYLATION.

Este gráfico indica que el gene set está enriquecido en los valores altos de la tabla, es decir, en los que se supone que están regulados al alza. En el archivo "HALLMARK_OXIDATIVE_PHOSPHORYLATION.html" también se puede observar una tabla con los genes que conforman el "leading edge" (indicados con un "yes" en la columna "core enrichment") se puede observar que estos genes son los que están situados en la parte superior de la tabla, lo cual corresponde con el hecho de que se trata de genes regulados al alza.

DPN unperturbed

Para acceder a la información de este fenotipo, se utilizó el archivo "gsea_report_for_na_neg_1711477851437.tsv" presente en el directorio "Apartado2/resultados/GSEA/my_analysis.GseaPreranked.1711477851437". En este caso, se identificaron 31 gene sets regulados al alza, de los cuales 19 resultaron significativamente enriquecidos (FDR < 0.25). Sin embargo, el FDR debería ser más estricto debido a que los gene sets han sido permutados. De esta forma, si se considera que el FDR ha de ser menor que 0.05 para que pueda considerarse que los gene sets han sido significativamente enriquecidos, hay 7 gene sets significativamente enriquecidos. Estos gene sets se muestran en la **Tabla 4**.

Tabla 4. Información del análisis de GSEA para los gene sets unperturbed.

Name	**NOM.p.val**	**FDR.q.val**
HALLMARK_EPITHELIAL_MESENCHYMAL_TRANSITION	0.000000000	0.000000000

HALLMARK_TNFA_SIGNALING_VIA_NFKB	0.000000000	0.009354908
HALLMARK_COAGULATION	0.000000000	0.020604870
HALLMARK_INFLAMMATORY_RESPONSE	0.000000000	0.026256757
HALLMARK_APICAL_SURFACE	0.014409222	0.034657646
HALLMARK_APICAL_JUNCTION	0.000000000	0.032965865
HALLMARK_TGF_BETA_SIGNALING	0.015151516	0.044705544

En el archivo “gsea_report_for_na_neg_1711477851437.html” presente en el mismo directorio se puede observar la tabla completa, así como acceder al apartado de “Detalles” de los gene sets para obtener más información.

A continuación se mostrarán los detalles del gene set “HALLMARK_EPITHELIAL_MESENCHYMAL_TRANSITION”, ya que la interpretación de los resultados es similar para los 7 gene sets y los gráficos obtenidos son similares.

Gene set HALLMARK_EPITHELIAL_MESENCHYMAL_TRANSITION

Los datos mostrados a continuación se han obtenido del archivo “HALLMARK_EPITHELIAL_MESENCHYMAL_TRANSITION.html” presente en el mismo directorio. A continuación (**Figura 16**) se muestran los gráficos característicos del análisis GSEA para el gene set HALLMARK_EPITHELIAL_MESENCHYMAL_TRANSITION.

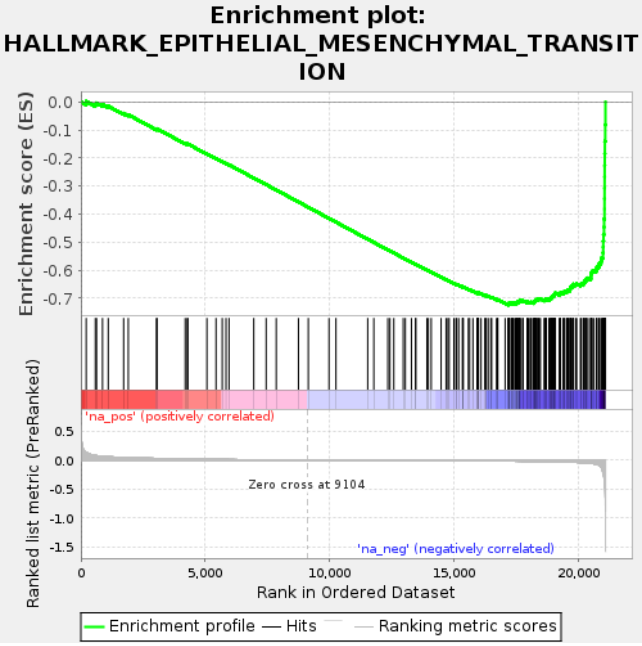

Figura 15. Gráfica de enriquecimiento para el gene set HALLMARK_EPITHELIAL_MESENCHYMAL_TRANSITION.

Este gráfico indica que el gene set está enriquecido en los valores bajos de la tabla, es decir, en los que se supone que están regulados a la baja. En el archivo "HALLMARK_EPITHELIAL_MESENCHYMAL_TRANSITION.html" también se puede observar una tabla con los genes que conforman el "leading edge" (indicados con un "yes" en la columna "core enrichment"). Se observa que estos genes son los que están situados en la parte inferior de la tabla, lo cual corresponde con el hecho de que se trata de genes regulados a la baja.

CONCLUSIONES

Con los datos preliminares se puede concluir que las diferencias de expresión génica no se deben a las diferencias en los distintos tratamientos, sino que se deben más a las diferencias individuales entre los pacientes. Además, el análisis con DESeq2 indica que hay una muy pequeña cantidad de genes diferencialmente expresados con ambos tratamientos a 24 horas. Además, en los heatmap se puede observar que las agrupaciones de genes no se deben a similitud en el tratamiento o en los tiempos sino a similitudes entre los pacientes (hecho que ya se confirmaba en los PCA realizados). La obtención de una cantidad tan baja de genes puede deberse a la eliminación del outlier, por lo que para mejorar los resultados habría que estudiar ampliar la muestra de forma que se tengan más de un dato por condición y paciente, que es lo que ocurre en este caso.

Además, tras realizar el GSEA se puede afirmar que el tratamiento DPN tiene un efecto significativo a lo largo del tiempo, ya que los genes pertenecientes al fenotipo "perturbed" son

los genes regulados al alza y los pertenecientes al fenotipo “unperturbed” son los genes regulados a la baja, y el enriquecimiento de estos genes concuerda con la parte alta y la parte baja de la tabla, respectivamente. Dicho de otra forma, los genes que se encuentran en la parte alta del archivo .rnk (zona de regulación al alza) se encuentran significativamente regulados al alza, y los genes que se encuentran en la parte baja del archivo .rnk (zona de regulación a la baja) se encuentran significativamente regulados a la baja, lo cual nos permite concluir que los cambios observados deberían ser más evidentes tras 48h, por lo que el DPN debería producir algún efecto en las primeras 24h.