

Análisis de expresión de RNA-seq para muestras de tiroides

Marina Ballesteros

6/14/2020

Contents

1 Abstract	2
2 Objetivos	2
3 Materiales	2
3.1 Diseño experimental	2
3.2 Datos	3
3.3 Software	3
4 Métodos: Procedimiento general del análisis (“Workflow”)	4
4.1 Definición de los datos	5
4.2 Filtración de los genes	5
4.3 Control de calidad de los datos filtrados	7
4.4 Normalización de los datos	11
4.5 Identificación de genes diferencialmente expresados	15
4.6 Anotación de los resultados	18
4.7 Búsqueda de patrones de expresión y agrupación de las muestras	20
4.8 Análisis de significación biológica	21
5 Resultados	21
5.1 Expresión diferencial	21
5.2 Búsqueda de patrones de expresión y agrupación de las muestras	25
5.3 Análisis de significación biológica	26
6 Discusión	28
7 Conclusión	29
8 Setup	29

1 Abstract

Este análisis se basa en el estudio de expresión de 30 muestras de RNA-seq pertenecientes al tejido tiroides con el fin de comparar los tres tipos de infiltración medio: not infiltrated tissues (NIT), small focal infiltrates (SFI) y extensive lymphoid infiltrates (ELI). Los datos han sido obtenidos gracias al proyecto The Genotype-Tissue Expression (GTEx), el cual es una base de datos pública para estudiar la expresión génica y la regulación de tejidos. En esta fuente de datos se recogen muestras de 54 tejidos no enfermo en aproximadamente 1000 individuos para llevar a cabo diferentes ensayos moleculares.

Los datos y el código completo del análisis se encuentran en el siguiente **repositorio github** [<https://github.com/marinabf93/Analisis-Expresion-RNAseq.git>]

2 Objetivos

Para este análisis he marcado dos objetivos bien diferenciados:

- a) Comparar los tres grupos entre sí e identificar los genes más significativos diferencialmente expresados en cada comparación con sus correspondientes anotaciones.
- b) Identificar los procesos biológicos, componentes celulares o funciones moleculares más afectados o implicados en el estudio.

3 Materiales

La razón de este trabajo es analizar bioinformáticamente los datos de un experimento con RNA-seq. Los datos en el que se ha basado el análisis han sido aportados junto al enunciado de la PEC pero pertenecen al portal GTEx (<https://www.gtexportal.org/home/>) (1)

Dentro de los distintos paquetes existentes para el análisis de RNA-seq, he escogido el **paquete edgeR** para realizar el análisis (2). El material en el que he basado todo mi análisis ha sido la guía de utilización del paquete edgeR: “edgeR: differential expression analysis of digital gene expression data” (<http://www.bioconductor.org/packages/release/bioc/vignettes/edgeR/inst/doc/edgeRUsersGuide.pdf>)

3.1 Diseño experimental

El **tipo de experimento** corresponde al análisis de RNA-seq, donde a través del diseño de un experimento se intenta responder a los objetivos planteados. Con el uso de la estadística y las diferentes herramientas bioinformáticas, se pretende procesar, analizar, visualizar y analizar los datos con el fin de responder a las cuestiones biológicas de partida.

El punto de partida de un experimento de RNA-Seq es un conjunto de muestras de RNA, típicamente asociadas con una variedad de condiciones de tratamiento. Cada muestra se secuencia, se asignan lecturas cortas al genoma apropiado y se registra el número de lecturas asignadas a cada característica genómica de interés. El conjunto de recuentos genéticos de cada muestra constituye la librería de expresión de esa muestra. El tamaño esperado de cada recuento es el producto del tamaño de la librería y la abundancia relativa de ese gen en esa muestra.

El paquete que he elegido (**edgeR**) trabaja en una tabla de recuentos de lectura de números enteros, con filas correspondientes a los genes y columnas a las librerías independientes. Los recuentos representan el

número total de lecturas alineadas a cada gen (u otro locus genómico). Esos recuentos pueden producirse a partir de lecturas alineadas por una variedad de herramientas de software de lectura corta.

Las lecturas pueden ser contadas de varias maneras. Cuando se realizan análisis a nivel genético, los conteos podrían ser para mapear lecturas en cualquier lugar del rango genómico del gen, como en este análisis, o los conteos podrían ser sólo para exones. Normalmente contamos lecturas que se superponen a cualquier exón para el gen dado, incluyendo el UTR como parte del primer exón.

3.2 Datos

En el caso particular de este análisis se parte directamente de los **datos de conteo** en forma de una tabla rectangular de valores enteros. La celda de la tabla en la fila g -ésima y la columna j -ésima de la tabla indica cuántas lecturas se han asignado al gen g en la muestra j .

Además de la tabla con los datos de conteo, se aporta una tabla llamada **targets** con toda la información necesaria relativa al estudio: el ID del experimento, el nombre de las muestras, el grupo al que pertenecen, el tipo de dato molecular, etc. Ambos datos han sido aportados por el profesor de la asignatura “Análisis de datos ómicos” de la UOC.

3.3 Software

Para comenzar el análisis se necesita instalar **R statistical software** el cual permite hacer análisis estadísticos, representaciones gráficas y lectura y creación de documentos en diferentes formatos. El software se puede descargar en la página web [<https://cran.r-project.org/index.html>] y solo deben seguirse las instrucciones indicadas en función del tipo de software del ordenador que se utilice para el análisis.

El análisis de RNA-seq que se presenta en este informe ha sido desarrollado con la versión 3.6.2 y todos los análisis se han llevado a cabo con la interfaz *RStudio*. Esta interfaz puede descargarse desde la página principal [<https://www.rstudio.com/>]

4 Métodos: Procedimiento general del análisis (“Workflow”)

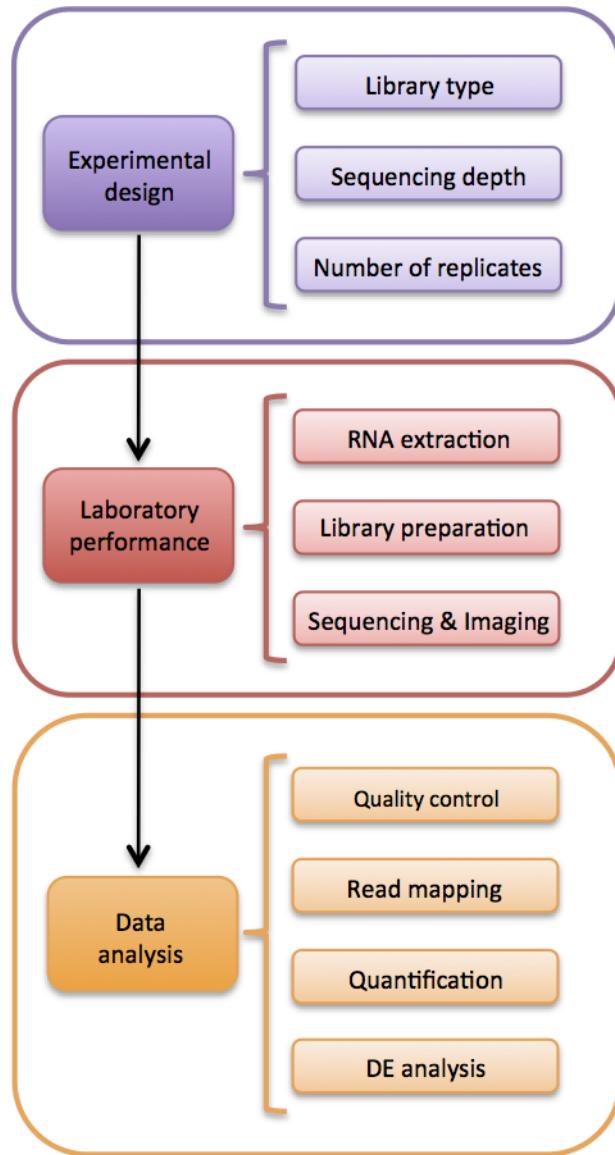


Figure 1: Workflow del análisis de RNA-seq.

A continuación resumiré de forma muy general los métodos utilizados en cada paso del flujo de trabajo. El desarrollo detallado de cada paso del análisis lo encontraréis en el archivo **Pipeline del análisis RNA-seq.Rmd** dentro del directorio principal del repositorio github indicado al inicio de este informe.

Antes de empezar con el análisis y a manejar la enorme cantidad de datos y ficheros que ello conlleva, crearé tres carpetas para la organización del mismo:

- La carpeta principal del análisis será “PEC2”, la cual también será mi directorio de trabajo.
- Una carpeta llamada **data** para almacenar todo tipo de datos del experimento y en los cuales basaré mi análisis. En esta carpeta guardaré los archivos *counts* y el archivo *targets*, en el cual se describirán los factores de estudio y sus niveles.
- En la carpeta **results** guardaré todos los resultados obtenidos en el análisis.

- La carpeta **figures** servirá para almacenar todo tipo de imágenes y figuras generadas durante el análisis.

4.1 Definición de los datos

El primer paso es leer los archivos **counts.csv** y **targets.csv** aportados junto al enunciado de la PEC, ambos archivos están ubicados en una la carpeta **data** en el directorio principal de trabajo.

El archivo *targets.csv* es un resumen de cada experiencia donde aparecen informaciones como el grupo, sexo o nombre de la muestra. La columna *group* de este archivo nos informa de los tres grupos de muestras existentes:

-Not infiltrated tissues (NIT) -Small focal infiltrates (SFI) -Extensive lymphoid infiltrates (ELI)

El archivo *counts.csv* contiene en sus columnas el nombre de las 292 muestras y cada fila pertenece a un gen diferente.

El siguiente paso fue escoger 30 muestras dentro de las 292 muestras posibles, de manera que tenga 10 muestras de cada uno de los tres grupos de tiroides. Para llevar a cabo esto, necesito seleccionar primero 10 muestras de cada grupo a través de la columna *group* del archivo **targets.csv**.

```
group_NIT<-subset(targets, Group=="NIT") [1:10,]
group_SFI<-subset(targets, Group=="SFI") [1:10,]
group_ELI<-subset(targets, Group=="ELI") [1:10,]

#Unimos los tres data frame creados en uno solo
targets_30<-Reduce(function(...) merge(...,all=TRUE), list(group_NIT, group_SFI, group_ELI))
```

Una vez he escogido las 30 muestras, extraigo las columnas correspondientes (en el archivo **counts.csv**) a esas 30 filas seleccionadas para finalmente obtener el archivo **counts_30**. Este archivo contiene las 30 muestras con 56202 observaciones (genes) y será la base de todo el análisis.

4.2 Filtración de los genes

Filtraré los genes reteniendo sólo aquellos genes que se expresen en todas las muestras y con un número mínimo de conteos. El proceso de filtraje es muy importante ya que los genes con recuentos muy bajos en las librerías no proporcionan apenas pruebas de expresión diferencial y, además, interfieren de forma negativa en algunas aproximaciones estadísticas. En consecuencia, estos genes con bajos conteos reducen el poder de detección de los genes de expresión diferencial y por ello es importante eliminarlos. Para llevar a cabo el filtraje necesito primero descargar el paquete **edgeR** de Bioconductor.

Existen diferentes métodos de filtrar los genes de baja expresión, pero antes debo transformar los conteos con el fin de tener en una misma escala todas las muestras y así evitar diferencias debido al distinto tamaño de las librerías.

En este estudio existen 10 réplicas biológicas por cada grupo, ya que el tamaño de muestra de cada grupo es 10. Por este motivo, estableceré un umbral mínimo de conteos por millón (CPM) en al menos 10 muestras para favorecer un filtraje donde los genes estén representados al menos una vez en todas las muestras en cada grupo.

El segundo umbral que he marcado para el filtraje es un mínimo de conteos por millón para cada gen. Para obtener el número de conteos por millón he utilizado la función **cpm** del paquete **edgeR**. Esta función permite convertir los conteos a CPMs, por lo tanto, se están normalizando los conteos para las diferentes profundidades de secuenciación en cada muestra.

Por regla general, se puede elegir un buen umbral mínimo de CPMs identificando el CPM que corresponde a un conteo de 10. Contrastando las tablas **counts_30** y **myCPM**, se observa que el umbral en este caso es

aproximadamente 0,15. Con la selección de genes que superen el umbral de 0,15 se obtiene una matriz lógica con los genes que han superado el umbral (TRUE) y los genes que están por debajo del umbral (FALSE). Se muestran los resultados de los tres primeros genes para la superación o no de dicho umbral.

```
##      GTEX.111CU.0226.SM.5GZXC GTEX.111FC.1026.SM.5GZX1 GTEX.111VG.0526.SM.5N9BW
## [1,] FALSE          FALSE          FALSE
## [2,] TRUE           TRUE           TRUE
## [3,] FALSE          FALSE          FALSE
##      GTEX.111YS.0726.SM.5GZY8 GTEX.11220.0226.SM.5N9DA GTEX.1128S.0126.SM.5H12S
## [1,] FALSE          FALSE          FALSE
## [2,] TRUE           TRUE           TRUE
## [3,] FALSE          FALSE          FALSE
##      GTEX.113JC.0126.SM.5EGJW GTEX.117XS.0526.SM.5987Q GTEX.117YW.0126.SM.5EGGN
## [1,] FALSE          FALSE          FALSE
## [2,] TRUE           TRUE           TRUE
## [3,] FALSE          FALSE          FALSE
##      GTEX.117YX.1226.SM.5H11S GTEX.1192W.0126.SM.5EGGS GTEX.1192X.1126.SM.5EGGU
## [1,] FALSE          FALSE          FALSE
## [2,] TRUE           TRUE           TRUE
## [3,] FALSE          FALSE          FALSE
##      GTEX.11DXY.0426.SM.5H12R GTEX.11EQ8.0826.SM.5N9FG GTEX.11EQ9.0626.SM.5A5K1
## [1,] FALSE          FALSE          FALSE
## [2,] TRUE           TRUE           TRUE
## [3,] FALSE          FALSE          FALSE
##      GTEX.11GS4.0826.SM.5986J GTEX.11NV4.0626.SM.5N9BR GTEX.11072.2326.SM.5BC7H
## [1,] FALSE          FALSE          FALSE
## [2,] TRUE           TRUE           TRUE
## [3,] FALSE          FALSE          FALSE
##      GTEX.11TUW.0226.SM.5LU8X GTEX.11XUK.0226.SM.5EQLW GTEX.1211K.0726.SM.5FQUW
## [1,] FALSE          FALSE          FALSE
## [2,] TRUE           TRUE           TRUE
## [3,] FALSE          FALSE          FALSE
##      GTEX.12584.0826.SM.5FQSK GTEX.12BJ1.0426.SM.5FQSO GTEX.13NZ9.1126.SM.5MR37
## [1,] FALSE          FALSE          FALSE
## [2,] TRUE           TRUE           TRUE
## [3,] FALSE          FALSE          FALSE
##      GTEX.13QJC.0826.SM.5RQKC GTEX.14ABY.0926.SM.5Q5DY GTEX.14AS3.0226.SM.5Q5B6
## [1,] FALSE          FALSE          FALSE
## [2,] TRUE           TRUE           TRUE
## [3,] FALSE          FALSE          FALSE
##      GTEX.14BMU.0226.SM.5S2QA GTEX.PLZ4.1226.SM.2I5FE GTEX.R55G.0726.SM.2TC6J
## [1,] FALSE          FALSE          TRUE
## [2,] TRUE           TRUE           TRUE
## [3,] FALSE          FALSE          FALSE
```

Una vez he filtrado los genes, hago un resumen con los genes que tienen un valor CPM superior al valor umbral. Dentro de los genes seleccionados como *TRUE*, seleccionaré los genes que tienen al menos 10 valores *TRUE* para cada gen; es decir, me quedaré con los genes que tengan representación en todas las muestras del grupo.

Los genes que superen los dos umbrales se seleccionan y recogen en el objeto *counts.keep* y estos serán los únicos conteos que conservaré para posteriores análisis. En este caso partimos de 56202 genes y tan solo conservaré 22990 genes tras la filtración.

```
##      Mode    FALSE    TRUE
```

```

## logical 33212 22990
## [1] 22990   30

```

Hasta aquí llega el proceso de filtraje, ahora debo convertir los conteos en un objeto de la clase **DEGList**; este objeto es propio del paquete *edgeR* y sirve para almacenar datos de conteos con los parámetros que se consideren pertinentes. En este caso sólo añadiré como parámetro la clasificación de las muestras en el grupo correspondiente gracias al objeto *group* creado junto con la tabla de conteos *counts_30*. El objeto *group* me permite identificar el grupo al que pertenece cada muestra tal y como se muestra en la siguiente tabla.

Table 1: Tabla del objeto DGEList donde para cada muestra se incluye el grupo, el tamaño de la librería y el factor de normalización.

	group	lib.size	norm.factors
GTEX.111CU.0226.SM.5GZXC	NIT	66132137	1
GTEX.111FC.1026.SM.5GZX1	NIT	55915131	1
GTEX.111VG.0526.SM.5N9BW	ELI	52167486	1
GTEX.111YS.0726.SM.5GZY8	NIT	48970410	1
GTEX.11220.0226.SM.5N9DA	NIT	53818958	1
GTEX.1128S.0126.SM.5H12S	NIT	47477039	1
GTEX.113JC.0126.SM.5EGJW	NIT	50137659	1
GTEX.117XS.0526.SM.5987Q	NIT	41440706	1
GTEX.117YW.0126.SM.5EGGN	SFI	37403355	1
GTEX.117YX.1226.SM.5H11S	NIT	52045327	1
GTEX.1192W.0126.SM.5EGGS	NIT	43547734	1
GTEX.1192X.1126.SM.5EGGU	NIT	43035398	1
GTEX.11DXY.0426.SM.5H12R	SFI	53058256	1
GTEX.11EQ8.0826.SM.5N9FG	SFI	55742090	1
GTEX.11EQ9.0626.SM.5A5K1	SFI	73420195	1
GTEX.11GS4.0826.SM.5986J	SFI	50356711	1
GTEX.11NV4.0626.SM.5N9BR	ELI	51917213	1
GTEX.11O72.2326.SM.5BC7H	SFI	68463803	1
GTEX.11TUW.0226.SM.5LU8X	SFI	44913853	1
GTEX.11XUK.0226.SM.5EQLW	ELI	49964100	1
GTEX.1211K.0726.SM.5FQUW	SFI	51857837	1
GTEX.12584.0826.SM.5FQSK	SFI	51981678	1
GTEX.12BJ1.0426.SM.5FQSO	SFI	54105277	1
GTEX.13NZ9.1126.SM.5MR37	ELI	61398921	1
GTEX.13QJC.0826.SM.5RQKC	ELI	48806596	1
GTEX.14ABY.0926.SM.5Q5DY	ELI	64676809	1
GTEX.14AS3.0226.SM.5Q5B6	ELI	41984061	1
GTEX.14BMU.0226.SM.5S2QA	ELI	44456272	1
GTEX.PLZ4.1226.SM.2I5FE	ELI	64405789	1
GTEX.R55G.0726.SM.2TC6J	ELI	15468340	1

4.3 Control de calidad de los datos filtrados

El objetivo del control de calidad es revelar posibles problemas técnicos u otros sesgos presentes en los datos. La mejor manera para llevar a cabo un control de calidad es de manera visual. El primer paso, por lo tanto, será representar los datos gráficamente.

El primer gráfico será una representación de los tamaños de las distintas librerías con un gráfico de barras

para ver si hay grandes discrepancias entre las muestras.

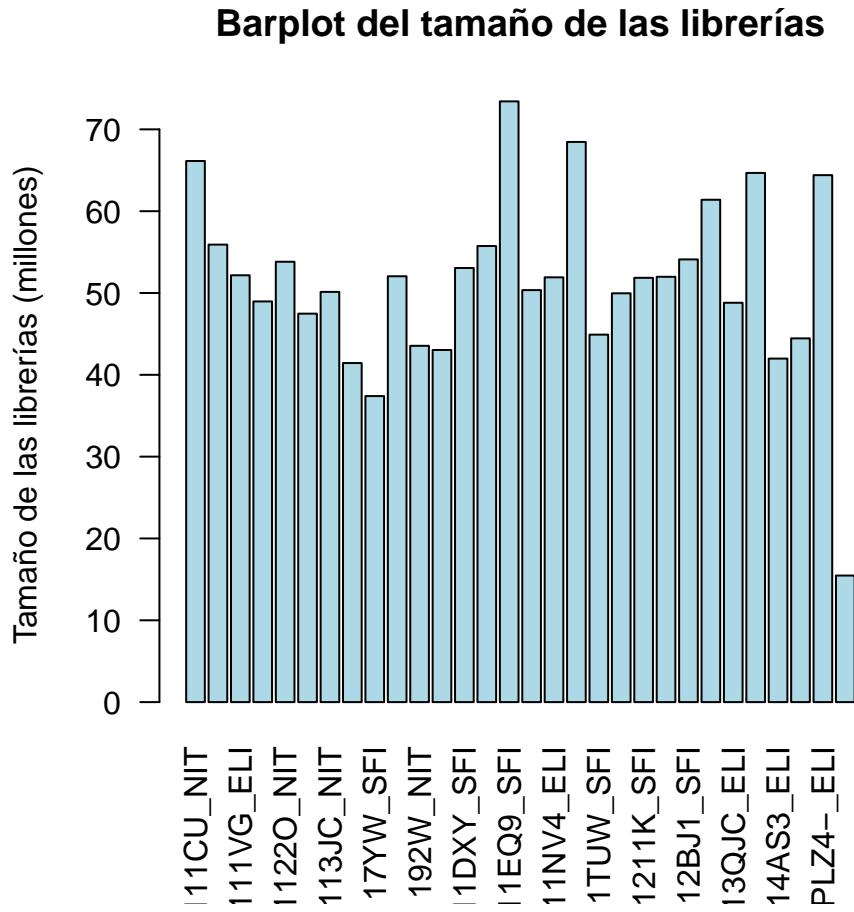


Figure 2: Barplot que representa el tamaño de las librerías de cada muestra.

En el gráfico se aprecia que las dimensiones de las librerías varían entre 15 y 70 millones de conteos. Estas enormes diferencias en el tamaño de las librerías indica que una normalización de los datos es necesaria antes de llevar a cabo los análisis de expresión diferencial.

Los datos de conteo no están distribuidos normalmente, así que si quiero examinar las distribuciones de los conteos en bruto necesito convertir los conteos en logaritmos. Una vez los conteos estén transformados en logaritmos, usaré gráficos de caja para comprobar la distribución de los recuentos leídos en la escala log2.

Boxplots de los logCPMs (no normalizados)

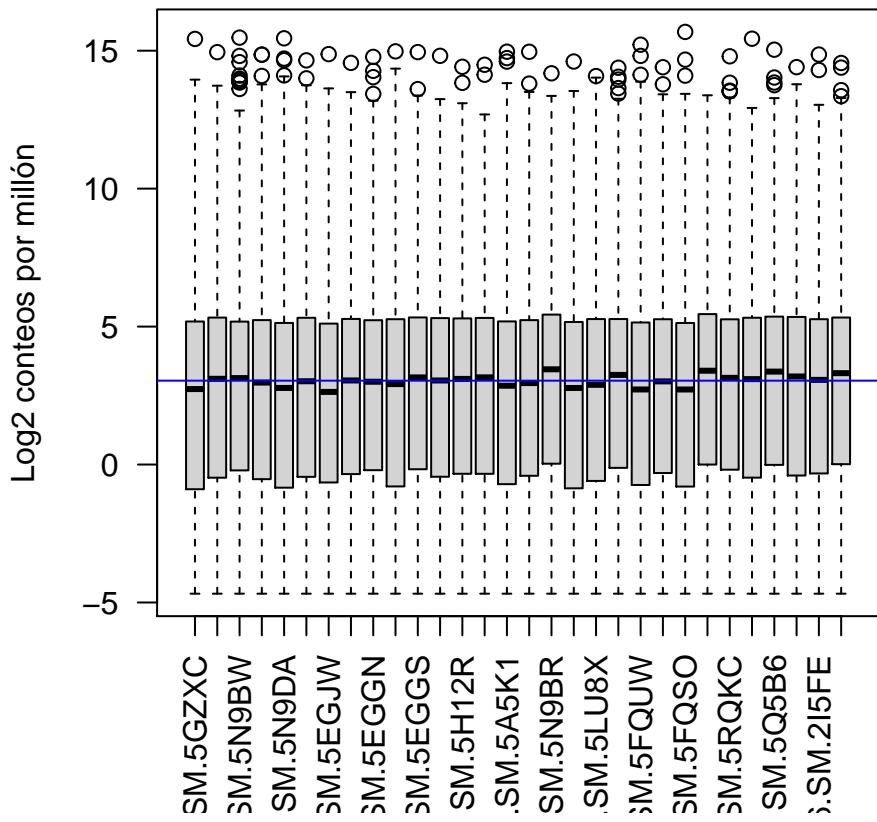


Figure 3: Boxplot de las dsitribuciones de las librerías.

De las 30 cajas, se observa que en general las distribuciones de densidad de las intensidades logarítmicas brutas no son idénticas pero aún así no muy diferentes. Si una muestra está realmente muy por encima o por debajo de la línea horizontal azul, se necesitaría entonces investigar esa muestra más a fondo. Los puntos dibujados más allá de los extremos de las cajas corresponden a valores outliers.

El siguiente gráfico que llevaré a cabo en este control de calidad visual es un **Mapa de color** gracias al paquete *mixOmics* (3). Este tipo de gráficos sirven para agrupar las muestras en base a algún método jerárquico. Por lo tanto, las muestras que se encuentren juntas serán las muestras más similares entre sí.

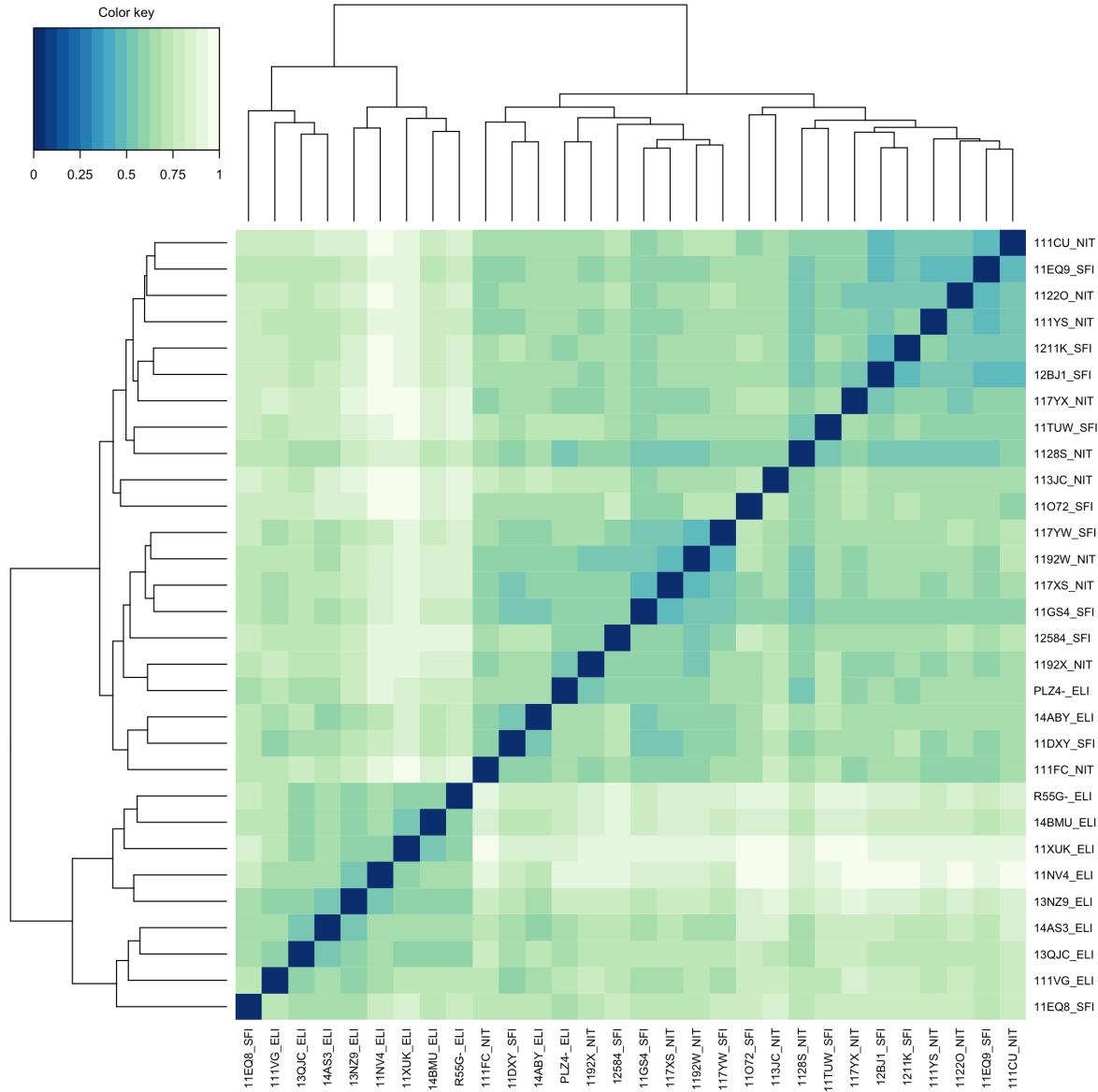


Figure 4: Mapa de color para cada muestra en cada grupo.

En el *Heatmap* no existe una clara agrupación entre los tres grupos, el grupo que más agrupado se encuentra es el *ELI* en la parte inferior del eje Y. Por el contrario, las muestras de los grupos *SFI* y *NIT* están entremezcladas. Según la escala de colores, el color azul representaría los genes que no han cambiado su expresión ; mientras que el color verde representa los genes que si han aumentado su expresión. Esto me hace indicar que en la inmensa mayoría de los casos, los genes de las diferentes muestras han sufrido un aumento en su expresión, veré si esta predicción se corresponde más adelante con el análisis de expresión diferencial.

Para acabar con este control de calidad visual, representaré un **Plot de componentes principales (PCA)**. Este tipo de gráficos es útil para visualizar el efecto global de las covariables experimentales y los efectos de los lotes. En este análisis, el plot PCA agrupa las muestras por grupos de genes que más significativamente han cambiado su expresión. Debido a que en este estudio existe un único factor con tres niveles: *SFI*, *NIT* y *ELI*; debería haber una clara separación de las muestras en función de estos tres niveles.

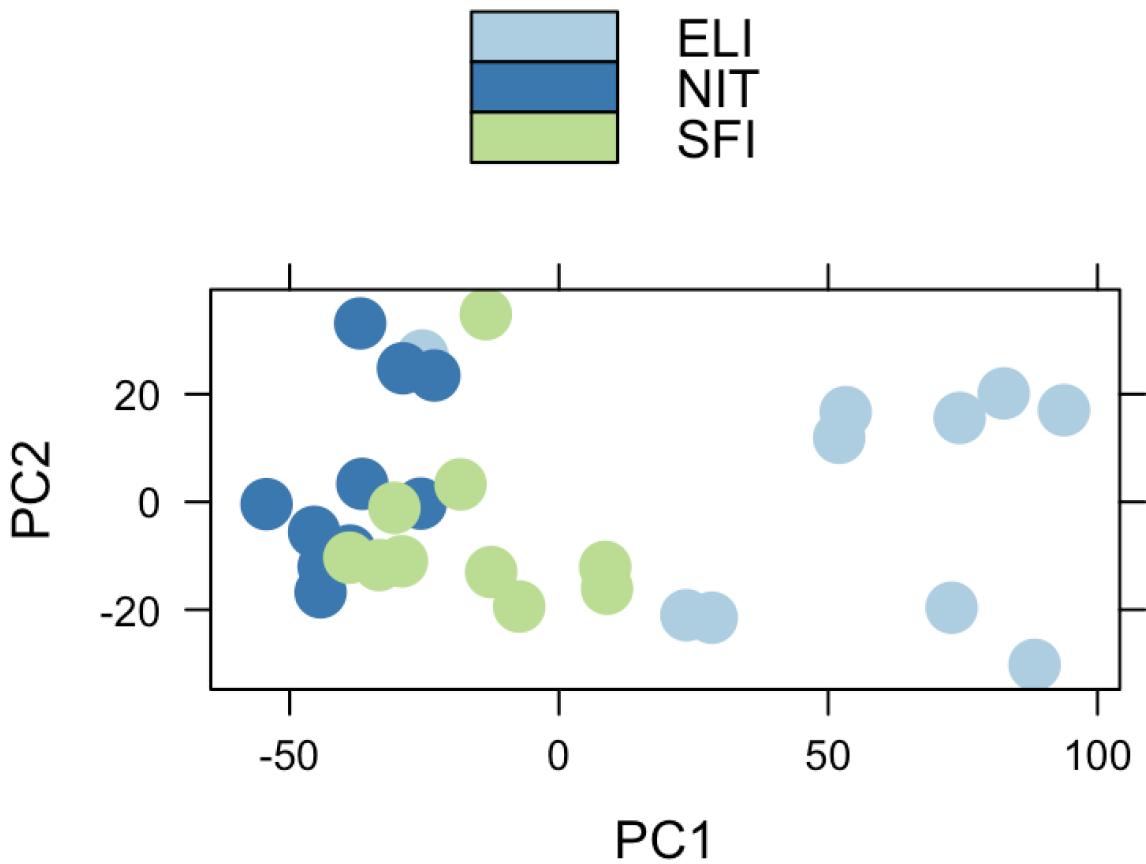


Figure 5: Plot PCA para el análisis de componentes principales.

En el gráfico de **análisis de componentes principales** se ve una clara separación de los tres grupos: las muestras pertenecientes al grupo *ELI* se sitúan a la derecha del gráfico, las muestras del grupo *SFI* están en la mitad baja del gráfico y, las muestras *NIT* se ubican en la parte derecha del plot. Sin embargo, se pueden observar dos muestras que no están situadas con su grupo: una muestra *SFI* y una muestra *ELI* se sitúan junto a las muestras del grupo *NIT*. Debido a que son sólo dos muestras, una de cada grupo, no habría de efectos de lote personalmente.

4.4 Normalización de los datos

Por último, procedo a la **normalización** de los datos filtrados. El paquete **edgeR** se ocupa del análisis de la expresión diferencial más que de la cuantificación de los niveles de expresión. Es decir, se ocupa de los cambios relativos en los niveles de expresión entre las condiciones. Por esta razón, los problemas de normalización se plantean sólo en la medida en que los factores técnicos tienen efectos específicos en la muestra.

Hay dos factores técnicos que pueden afectar a los recuentos de lectura de expresión diferencial:

La profundidad de secuenciación de cada muestra de ARN. El paquete *edgeR* ajusta automáticamente cualquier análisis de expresión diferencial para variar las profundidades de secuenciación, representadas por diferentes tamaños de librería. **La producción total de ARN por célula.** Esto suele ser importante cuando un pequeño número de genes se expresan en gran medida en una muestra, pero no en otra. Este efecto provoca que los genes altamente expresados consuman una proporción sustancial del tamaño total de la librería, lo que hace que los genes restantes no estén suficientemente muestreados en esa muestra.

Como el primer factor se corrige automáticamente con el paquete *edgeR*, debo corregir el segundo factor a través de la función **calcNormFactors**. Esta función encontrará un conjunto de factores de escala para los tamaños de las bibliotecas con el fin de minimizar los cambios de pliego entre las muestras para la mayoría de genes. El método por defecto para calcular estos factores de escala utiliza una media recortada de valores M (TMM) entre cada par de muestras. La multiplicación del tamaño original de la librería por el factor de escala se llamará el **tamaño efectivo de la librería**, que sustituye al tamaño original de la librería en todos los análisis.(4)

En este caso he usado el método por defecto, Trimmed Mean of M-values (TMM); ya que observando un poco los datos del archivo *counts_30* se ve que el recuento total de lecturas depende en gran medida de unas pocas transcripciones altamente expresadas.

Table 2: Tabla del objeto DGEList tras el proceso de normalización.

	group	lib.size	norm.factors
GTEX.111CU.0226.SM.5GZXC	NIT	66132137	0.8822941
GTEX.111FC.1026.SM.5GZX1	NIT	55915131	1.0484240
GTEX.111VG.0526.SM.5N9BW	ELI	52167486	0.9654989
GTEX.111YS.0726.SM.5GZY8	NIT	48970410	0.9651440
GTEX.1122O.0226.SM.5N9DA	NIT	53818958	0.8689757
GTEX.1128S.0126.SM.5H12S	NIT	47477039	1.0289663
GTEX.113JC.0126.SM.5EGJW	NIT	50137659	0.8861010
GTEX.117XS.0526.SM.5987Q	NIT	41440706	1.0196200
GTEX.117YW.0126.SM.5EGGN	SFI	37403355	1.0377109
GTEX.117YX.1226.SM.5H11S	NIT	52045327	0.9629562
GTEX.1192W.0126.SM.5EGGS	NIT	43547734	1.1006338
GTEX.1192X.1126.SM.5EGGU	NIT	43035398	1.0297697
GTEX.11DXY.0426.SM.5H12R	SFI	53058256	1.0360441
GTEX.11EQ8.0826.SM.5N9FG	SFI	55742090	1.0359237
GTEX.11EQ9.0626.SM.5A5K1	SFI	73420195	0.9274279
GTEX.11GS4.0826.SM.5986J	SFI	50356711	1.0094869
GTEX.11NV4.0626.SM.5N9BR	ELI	51917213	1.1269897
GTEX.11O72.2326.SM.5BC7H	SFI	68463803	0.8734407
GTEX.11TUW.0226.SM.5LU8X	SFI	44913853	0.9799702
GTEX.11XUK.0226.SM.5EQLW	ELI	49964100	0.9998036
GTEX.1211K.0726.SM.5FQUW	SFI	51857837	0.9045863
GTEX.12584.0826.SM.5FQSK	SFI	51981678	1.0356130
GTEX.12BJ1.0426.SM.5FQSO	SFI	54105277	0.8834430
GTEX.13NZ9.1126.SM.5MR37	ELI	61398921	1.1560360
GTEX.13QJC.0826.SM.5RQKC	ELI	48806596	1.0291883
GTEX.14ABY.0926.SM.5Q5DY	ELI	64676809	1.0313566
GTEX.14AS3.0226.SM.5Q5B6	ELI	41984061	1.1304005
GTEX.14BMU.0226.SM.5S2QA	ELI	44456272	1.0074671
GTEX.PLZ4.1226.SM.2I5FE	ELI	64405789	1.0387770
GTEX.R55G.0726.SM.2TC6J	ELI	15468340	1.0890683

Puede observarse que los factores de normalización han cambiado tras la normalización. Un factor de normalización inferior a uno indica que un pequeño número de genes de alto recuento monopoliza la secuencia, lo que hace que los recuentos de otros genes sean inferiores a lo que sería habitual dado el tamaño de la librería. Como resultado, el tamaño de la librería se reducirá, de forma análoga a la escalada de los recuentos al alza en esa librería. Por el contrario, un factor superior a uno aumenta el tamaño de la librería y equivale a reducir los recuentos.

El rendimiento del procedimiento de normalización de la TMM puede examinarse mediante gráficos de diferencia media o **plot MD**. Estos gráficos muestran la expresión media (media: eje x) frente a los cambios de logaritmo (diferencia: eje y). Para visualizar la diferencia de los gráficos antes y después de la normalización, he generado cuatro **plots MD** donde se representan las muestras *GTEX.111CU.0226.SM.5GZXC* y *GTEX.13NZ9.1126.SM.5MR37*, dos plots para cada muestra. He elegido la librería de la muestra *GTEX.111CU.0226.SM.5GZXC* porque es una de las muestras con el factor de normalización más pequeño: 0.882. Para tener el ejemplo contrario, he representado igualmente la muestra *GTEX.13NZ9.1126.SM.5MR37* la cual tiene uno de los factores de normalización más elevados: 1.156.

Estos son los gráficos de ambas muestras **antes** de la normalización.

GTEX.111CU.0226.SM.5GZXC GTEX.13NZ9.1126.SM.5MR37

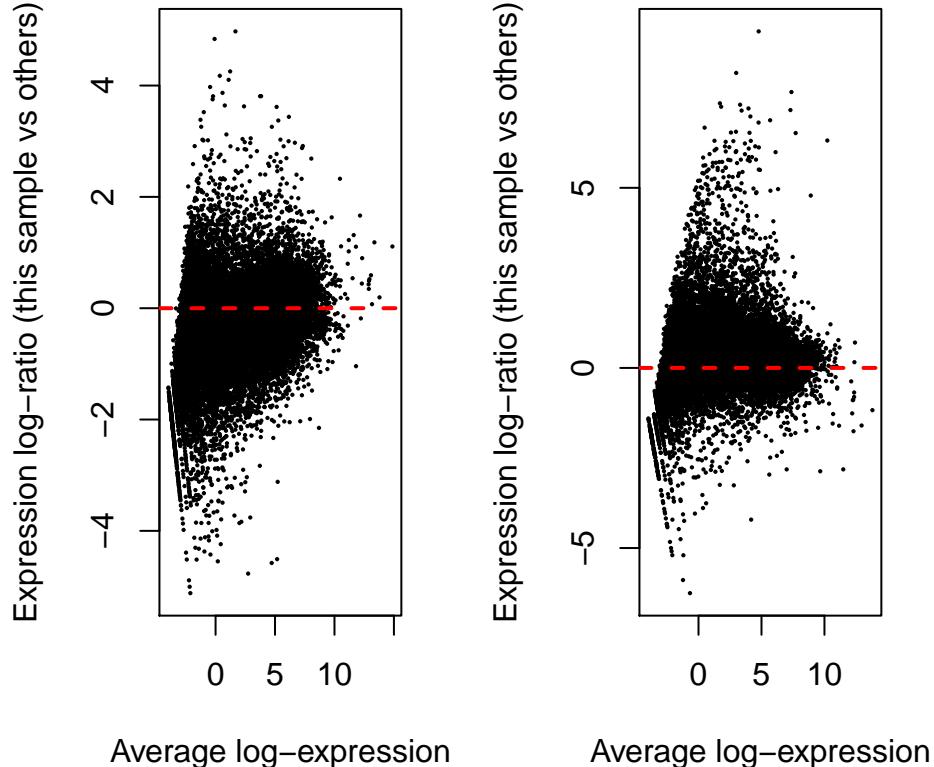
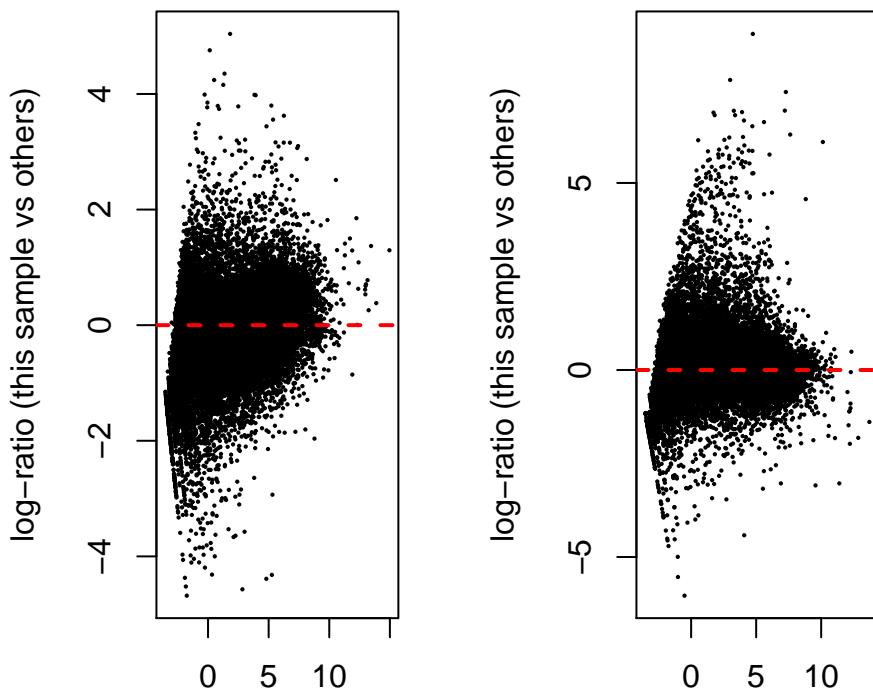


Figure 6: Plot MD para el logaritmo de los conteos, normalizados sólo para el tamaño de las librerías.

En ambos casos se ve con claridad que los valores no se sitúan en torno a un cambio de pliegue de cero marcado por la línea roja; sino que las muestras están por debajo y por encima de un cambio de pliegue de cero respectivamente.

Sin embargo, en los siguientes *plot MD* se aprecia que el grueso de genes para ambas muestras se sitúan en torno a un cambio de pliegue en torno a cero, por lo que el sesgo se ha corregido con la normalización del *objeto y*.

GTEX.111CU.0226.SM.5GZ GTEX.13NZ9.1126.SM.5MF



verage log CPM (this sample and verage log CPM (this sample and

Figure 7: Plot MD para el objeto y, normalizado tanto para el tamaño de las librerías como para el sesgo de composición.

Otro gráfico que se suele representar tras la normalización de los datos, es un **plot MDS** donde se representan las similitudes relativas de las 30 muestras en función del tipo de infiltración existente en el tiroides. Las similitudes se cuantifican en forma de cambios de pliegue (log fold change) entre muestras.

En el *plot MDS* la primera dimensión representa la magnitud del cambio biológico que mejor separa las muestras y, por lo tanto, el cambio biológico que representa la mayor proporción de variación de los datos.

Plot MDS para los tres grupos de infiltración

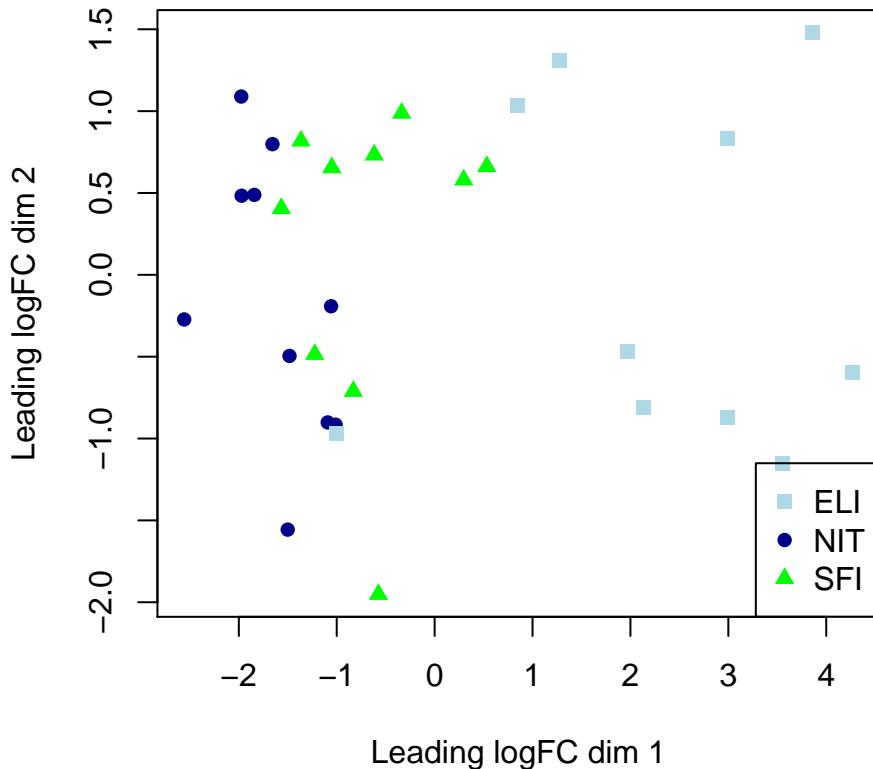


Figure 8: Plot MDS para el análisis de distancias entre muestras

Existe una clara agrupación entre los tres grupos, lo que muestra un efecto según el tipo de infiltración. Esta tendencia era de esperar puesto que en este análisis es el único factor a tener en cuenta y puesto que el plotMDS es un plot muy similar al plot PCA, donde ya se obtuvieron los mismos resultados.

4.5 Identificación de genes diferencialmente expresados

4.5.1 Matriz de diseño

La matriz de diseño contiene los predictores para cada muestra; en la matriz se asigna un coeficiente a cada grupo. La matriz de diseño se crea a partir del *objeto y* ya que contiene la información sobre el grupo experimental al que pertenece cada muestra. En este análisis, lo interesante son las diferencias entre los tres tipos de infiltración. Por lo tanto, creo una matriz de diseño utilizando el factor *tipo de infiltración*.

Table 3: Matriz de diseño basada en el factor tipo de infiltración.

	ELI	NIT	SFI
111CU_NIT	0	1	0
111FC_NIT	0	1	0
111VG_ELI	1	0	0
111YS_NIT	0	1	0
1122O_NIT	0	1	0
1128S_NIT	0	1	0

	ELI	NIT	SFI
113JC_NIT	0	1	0
117XS_NIT	0	1	0
117YW_SFI	0	0	1
117YX_NIT	0	1	0
1192W_NIT	0	1	0
1192X_NIT	0	1	0
11DXY_SFI	0	0	1
11EQ8_SFI	0	0	1
11EQ9_SFI	0	0	1
11GS4_SFI	0	0	1
11NV4_ELI	1	0	0
11O72_SFI	0	0	1
11TUV_SFI	0	0	1
11XUK_ELI	1	0	0
1211K_SFI	0	0	1
12584_SFI	0	0	1
12BJ1_SFI	0	0	1
13NZ9_ELI	1	0	0
13QJC_ELI	1	0	0
14ABY_ELI	1	0	0
14AS3_ELI	1	0	0
14BMU_ELI	1	0	0
PLZ4-_ELI	1	0	0
R55G-_ELI	1	0	0

4.5.2 Estimación de la dispersión

El modelo probabilístico **Binomial Negativo** es el elegido por el paquete *edgeR* para modelar los datos de conteo. El primer paso ha sido entonces estimar la dispersión de cada tránskrito a partir de la variabilidad total para todos los genes.

Para experimentos con un solo factor, como en este caso (tipo de infiltración) donde se buscan comparaciones por pares entre grupos, *edgeR* utiliza el método de máxima probabilidad condicional ajustada por cuantiles (qCML). El **método qCML** es un enfoque clásico de este paquete donde se calcula la probabilidad condicionándose a los recuentos totales de cada etiqueta y utiliza pseudocuentas después de ajustar los tamaños de las librerías. Es decir, primero se estima la dispersión común o variabilidad total (llamada *common dispersión*) y luego se estima la dispersión gen a gen (denominada *dispersión Tagwise*).

La estimación de la dispersión devuelve el objeto *DGEList* con entradas adicionales para las dispersiones NB estimadas para todos los genes, como es el **coeficiente de variación biológica (BCV)** de cada gen. En este caso he obtenido una dispersión común de 0.24 y el coeficiente BCV se ha calculado a través de la raíz cuadrada de esta dispersión, obteniendo un coeficiente BCV de 48.9%. El coeficiente BCV se ha representado gráficamente junto a las dispersiones individuales de cada gen, de este modo puede comprobarse si la dispersión común representa realmente la dispersión existente entre los genes. Estos gráficos se obtienen al representar la raíz cuadrada de las dispersiones estimadas frente al logaritmo en base 2 de las lecturas por millón.

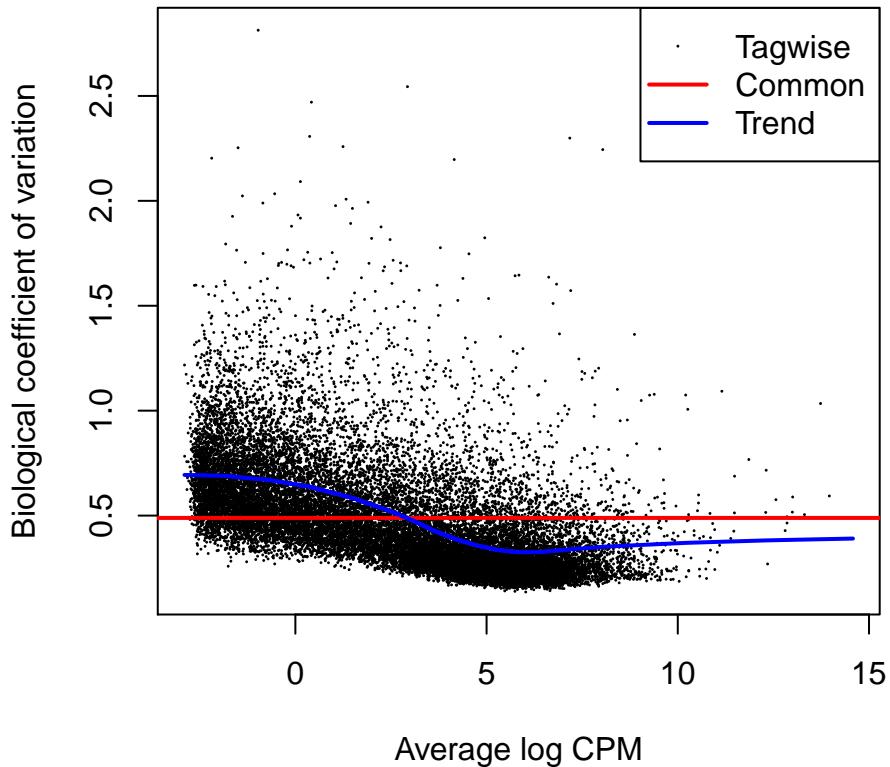


Figure 9: Plot BCV para el cálculo del coeficiente de variación biológica (BCV) y el cálculo de la dispersión entre las muestras.

El gráfico *BCV* representa las dispersiones estimadas gen a gen (dispersión Tagwise) a partir de la dispersión común, representada por la línea roja. Cada punto en el gráfico es un gen y la línea azul dibuja la tendencia de variación biológica con el aumento de los conteos. De manera general, un valor óptimo del coeficiente de variación biológico entre 0.2 y 0.4 favorecería la detección de genes diferencialmente expresados; en este caso el coeficiente es de 0.49, ligeramente superior al valor óptimo. Esto podría llevarme posteriormente a seleccionar menos genes DE de los que realmente hay.

4.5.3 Análisis de expresión diferencial

El **análisis de expresión diferencial** estima las dispersiones de quasi-probabilidad (QL) alrededor de la tendencia de dispersión usando la función *glmQLFit*. Esta función devuelve un objeto DGEGLM al que llamo **fit** que contiene los valores estimados de los coeficientes GLM para cada gen, así como la tendencia de dispersión de la media ajustada de QL, las estimaciones de QL ajustadas y los grados de libertad previos (df). El objetivo del análisis de expresión diferencial se identifican los genes que son atípicos de la tendencia de dispersión media NB.

En la siguiente tabla se muestran los valores del análisis de expresión diferencial de los 5 primeros genes.

Table 4: Resultado del análisis de expresión diferencial para los cinco primeros genes en cada grupo.

	ELI	NIT	SFI
2	-11.29316	-11.33698	-11.32599
8	-15.68217	-15.64372	-16.21529
9	-13.68783	-15.10299	-15.16598

	ELI	NIT	SFI
10	-11.26547	-10.93251	-11.73538
12	-15.65668	-15.10616	-15.83512

Los resultados visuales se muestran a continuación con el *plot QLDisp*.

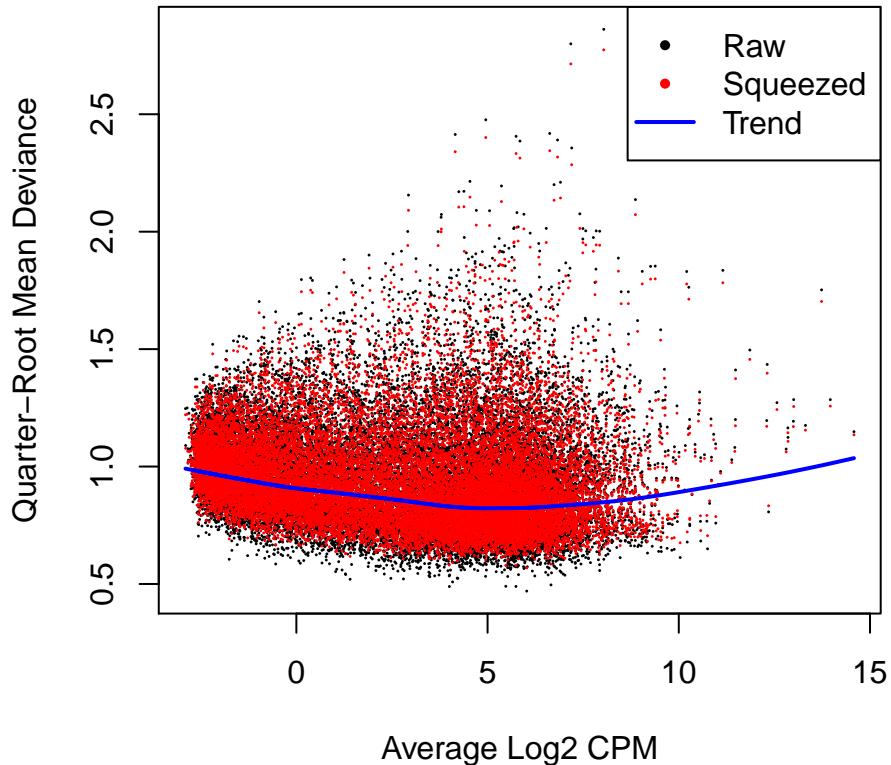


Figure 10: Plot QLDisp para el cálculo genes atípicos a la tendencia de dispersión media binomial negativa.

En este estudio hay un interés en comparar los tres tipos de grupos entre sí, por lo que realicé **tres contrastes** para continuar con el análisis de expresión diferencial:

- SFI vs NIT
- ELI vs NIT
- ELI vs SFI

En el paquete *edgeR* la función **exactTest** realizó el test exacto binomial negativo y la función **topTags** me devolvió una tabla con los genes DE más significativos para cada contraste. Por lo tanto, en esta etapa del análisis los datos de entrada fueron el *objeto y* y los datos de salida una tabla con la lista de genes diferencialmente expresados en cada contraste. En el apartado *Resultados* de este informe se muestran los resultados obtenidos.

4.6 Anotación de los resultados

El proceso de anotación consiste en relacionar los identificadores *ENTREZID* de la primera columna de las tablas *topTags* con información más fácil de manejar como el *Gene Symbol* o *Gene Name*. Las tablas *topTags* se obtuvieron en el análisis de expresión diferencial y se encuentran en el apartado “Resultados” en

este informe. El símbolo del gen y su descripción se añadirán a los resultados de las tablas gracias al paquete de anotación del genoma humano **org.Hs.eg.db**, el cual permite asociar los identificadores *ENTREZID* con el nombre y la descripción de los genes.

El resultado de las anotaciones son tres archivos excel registrados en el directorio **resultados** del proyecto que se indica en la dirección de github proporcionada al inicio del informe bajo los nombres de: anotaciones_ELIvsNIT.csv, anotaciones_SFIvsNIT.csv y anotaciones_ELIvsSFI.csv.

Los resultados de las tablas de selección junto a las anotaciones se pueden visualizar a través de un gráfico **volcano plot**. En este gráfico se representa en el eje de abscisas los cambios de expresión en escala logarítmica; mientras que en el eje de ordenadas se representa la significancia del gen a través del estadístico B en escala logarítmica. Yo he representado tres volcanos, uno para cada contraste y en cada uno de ellos, he representado los cuatro primeros genes más diferencialmente expresados de cada tabla. Los tres volcanos se recogen en el archivo *Volvanos.pdf* del directorio **figures** del repositorio github del análisis, pero a continuación se muestra como ejemplo el volcano correspondiente al primer contraste.

Genes diferencialmente expresados en ELIvsNIT

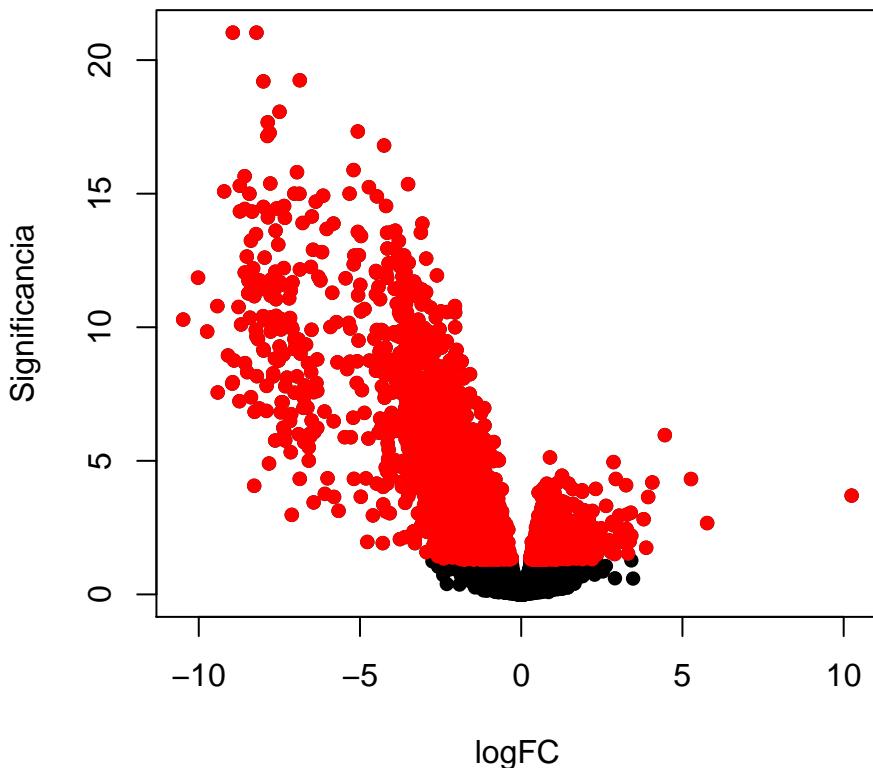


Figure 11: Volcanoplot para el contraste ELI vs NIT.

Para reforzar la visualización gráfica de la anotación, he representado el gen DE más significativo de cada lista para el que exista anotación y así poder verificar los niveles de expresión de este gen concreto en cada grupo. Para mirar la expresión conjunta de cada gen seleccionado en los tres grupos he utilizado un gráfico de barras con los valores de expresión logarítmica normalizados en el objeto **y** (**y\$counts**).

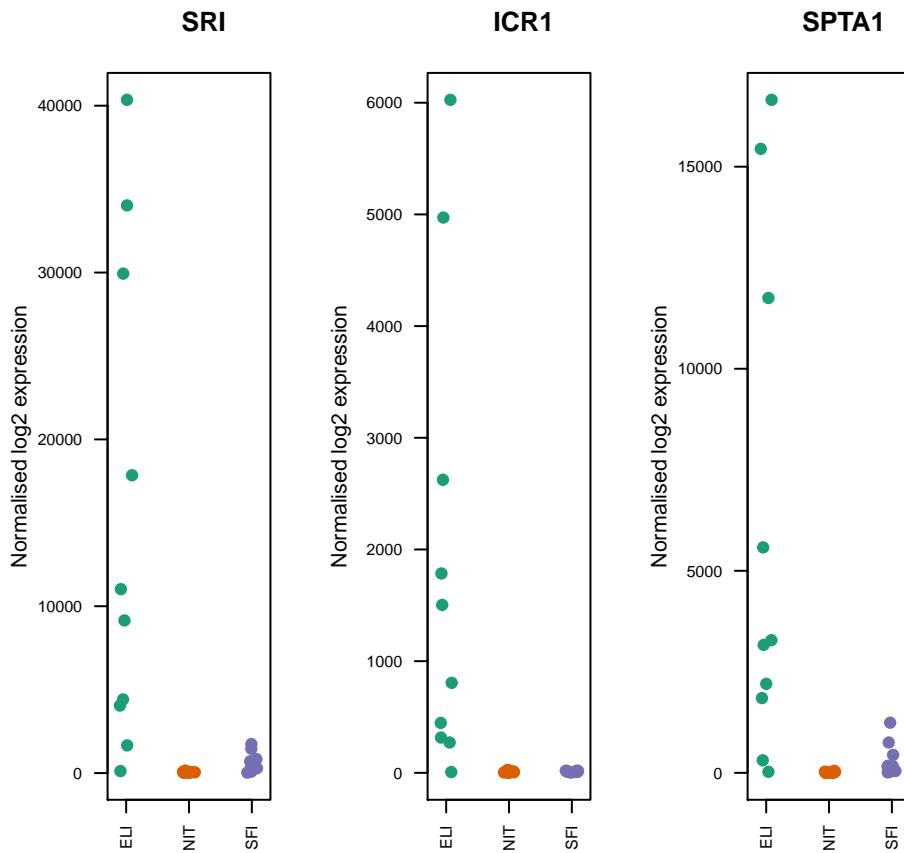


Figure 12: Gráfico de barras de la expresión individual de los genes SRI, ICR1 y SPTA1.

El resultado común en todos los contrastes es que los tres genes analizados de manera individual tiene su mayor nivel de expresión en el **grupo ELI** que corresponde con *extensos filtrados linfoides* y la expresión más moderada se encuentra en el grupo **NIT** que representa *tejidos no infiltrados*.

4.7 Búsqueda de patrones de expresión y agrupación de las muestras

Para poder llevar a cabo un contraste entre más de dos grupos, he recurrido al **análisis unidireccional de varianza (ANOVA)** para cada gen. Para ello, primero creé una matriz de contrastes a partir de la función *makeContrasts*, donde cada columna representa un contraste entre dos grupos de interés.

Table 5: Matriz de contrastes.

	SFIvsNIT	ELIvsNIT	ELIvsSFI
ELI	0	1	1
NIT	-1	-1	0
SFI	1	0	-1

La función *glmQLFTest* lleva a cabo la prueba de cuasi-probabilidad (QL) del test F que se aplica para identificar los genes que se expresan diferencialmente entre los tres grupos. Esto combina las tres comparaciones por pares en un solo estadístico F y un p-valor. En este caso el input es el objeto *fit* y como output se obtiene una lista con los genes DE más significativos en los tres contrastes.

4.8 Análisis de significación biológica

El análisis de enriquecimiento biológico (GEA) sirve para interpretar los resultados de la expresión diferencial en un contexto biológico. Su objetivo es identificar las funciones, rutas metabólicas o procesos biológicos donde intervienen los genes más diferencialmente expresados.

En este caso, se identificarán los términos GO que están sobrerepresentados en los tres grupos: ELI, SFI y NIT. el input u objeto de entrada será el objeto *fit* generado en la estimación de la dispersión y el output será una tabla donde se listan las vías, procesos y componentes celulares más afectados.

5 Resultados

5.1 Expresión diferencial

Para el análisis de expresión diferencial se han comparado todos los grupos entre sí, por lo que en total he realizado 3 contrastes. Para cada uno de los tres contrastes se han obtenido como resultados:

- Tabla topTags con los genes diferencialmente expresados más significativos.
- Resumen de los genes que han cambiado su expresión: upregulated o downregulated y genes que no han cambiado su expresión
- PlotSmear donde se grafican los genes y los cambios de pliegue de los mismos

El gráfico **plotSmear** se basa en la representación del logaritmo en base 2 del cambio de pliegue (logFC en el eje Y) correspondiente a cada gen frente a la media del logaritmo en base 2 de los conteos por millón (Average logCPM en el eje X). Los puntos que representan los genes se colorearán de rojo si el p-valor ajustado es inferior a 0.01; es decir, los puntos rojos representan los genes diferencialmente expresados. Las líneas azules horizontales muestran los cambios cuádruples de pliegue; en consecuencia, los genes fuera de ambas líneas serán los genes que más cambien su expresión.

Una vez explicados los tres tipos de resultados, los presento para cada contraste por separado.

5.1.1 Contraste ELI vs NIT

Realización del test exacto negativo y obtención de la tabla con los genes DE más significativos.

```
## Comparison of groups: NIT-ELI
##      logFC      logCPM      PValue          FDR
## 49043 -8.943159 5.067141 4.969077e-26 9.354330e-22
## 42101 -8.209736 4.051008 8.137738e-26 9.354330e-22
## 48230 -6.866932 3.576271 7.413892e-24 5.681513e-20
## 16760 -7.996948 4.395562 1.085000e-23 6.236039e-20
## 31449 -7.495456 6.112780 1.868759e-22 8.592552e-19
## 6717  -7.860853 6.669954 5.567370e-22 2.133231e-18
## 45681 -5.066702 3.887904 1.425218e-21 4.680825e-18
## 3388  -7.802553 3.769604 1.839405e-21 5.285989e-18
## 3379  -7.871562 4.427982 2.702476e-21 6.903325e-18
## 49407 -4.252927 5.358100 6.820687e-21 1.568076e-17
```

El siguiente paso fue seleccionar los genes DE para el contraste ELI vs NIT. Se obtuvo como resultado un total de 2748 genes con un valor FDR inferior al nivel de significación ($\alpha = 0.05$). De estos genes DE, 1954 disminuyen su expresión y 794 están sobreexpresados.

```

##          NIT-ELI
## Down      1954
## NotSig   20242
## Up       794

```

Y, por último, la representación gráfica de los resultados con el **plotSmear**.

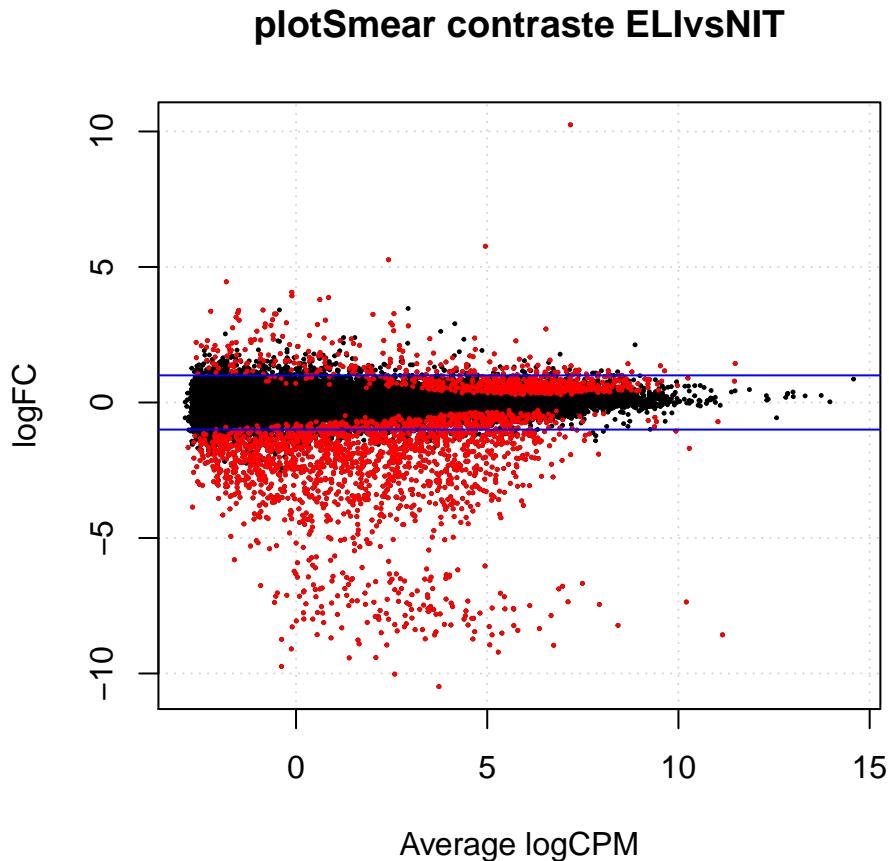


Figure 13: PlotSmear para el contraste ELI vs NIT.

5.1.2 Contraste SFI vs NIT

Realización del test exacto negativo y obtención de la tabla *topTags*.

```

## Comparison of groups: NIT-SFI
##      logFC      logCPM      PValue        FDR
## 39069 -3.945330  5.463306 2.741316e-07 0.002617619
## 16760 -3.306538  4.395562 3.692230e-07 0.002617619
## 39075 -4.095585 11.152286 4.273701e-07 0.002617619
## 3379  -3.446150  4.427982 4.554361e-07 0.002617619
## 39245 -4.090456  5.289069 8.636421e-07 0.003971026
## 6708  -4.192699  5.384217 1.351174e-06 0.004304912
## 6717  -3.182455  6.669954 1.409802e-06 0.004304912
## 6736  -4.351847  4.673461 1.645447e-06 0.004304912
## 6694  -3.761375  4.625580 1.685264e-06 0.004304912
## 39200 -4.242854  1.195602 2.082301e-06 0.004614124

```

Selección de genes DE para el contraste SFI vs NIT. Para este contraste apenas se encuentran genes DE, tan sólo 41 genes de los cuales 39 disminuyen su expresión y 2 la aumentan.

```
##          NIT-SFI
## Down      39
## NotSig   22949
## Up       2
```

Representación gráfica con **plotSmear**. Al haber tan pocos genes DE con respecto al total, los puntos rojos apenas se aprecian visualmente.

plotSmear **contraste SFIvsNIT**

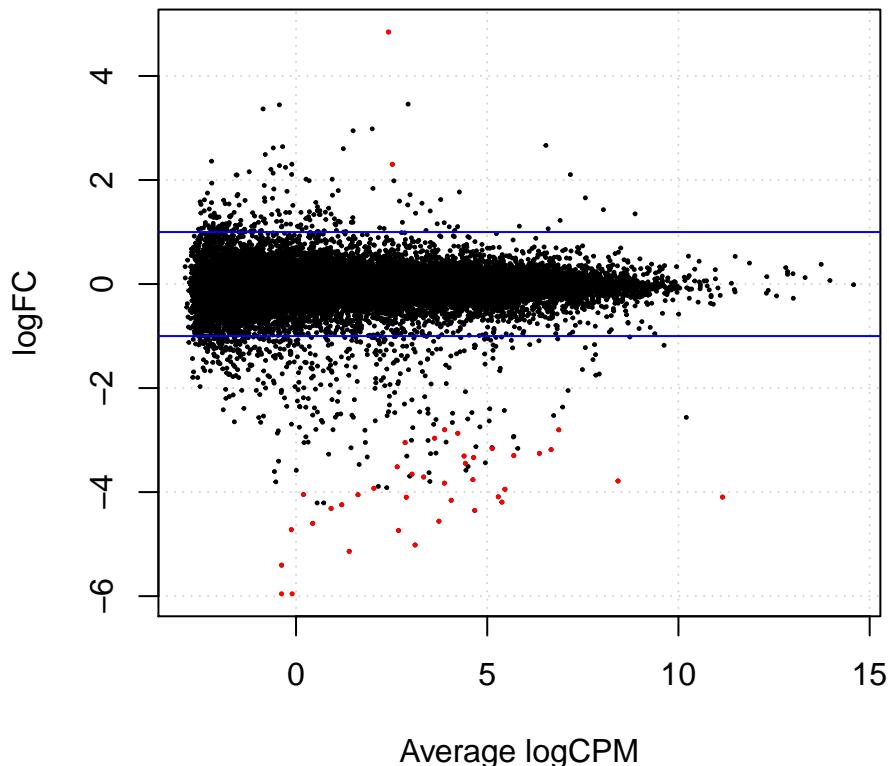


Figure 14: PlotSmear para el contraste SFI vs NIT.

5.1.3 Contraste ELI vs SFI

Realización del test exacto negativo y obtención de la tabla con los genes DE.

```
## Comparison of groups: SFI-ELI
##      logFC    logCPM      PValue        FDR
## 48230 -6.133354 3.576271 1.127308e-20 1.349124e-16
## 42101 -6.826769 4.051008 1.173661e-20 1.349124e-16
## 3388  -7.341381 3.769604 6.474492e-20 4.961619e-16
## 31449 -6.760816 6.112780 9.576896e-20 5.504321e-16
## 45890 -3.481509 2.640516 4.703517e-19 2.162677e-15
## 45681 -4.551486 3.887904 1.469646e-18 5.631193e-15
```

```

## 49407 -3.887053 5.358100 2.791898e-18 9.169392e-15
## 10926 -4.177496 4.624797 4.267589e-18 1.226398e-14
## 49043 -6.545052 5.067141 9.304891e-18 2.376883e-14
## 49026 -4.041558 1.425183 2.815890e-17 6.286894e-14

```

Selección de genes DE para el contraste ELI vs SFI. Para este contraste un total de 2470 genes tiene un FDR inferior al nivel de significación. De ellos 686 aumentan su expresión y 1784 la disminuyen.

```

##           SFI-ELI
## Down      1784
## NotSig   20520
## Up       686

```

Representación gráfica con **plotSmear**. En este contraste la mayoría de los puntos rojos están bajo un logFC negativo en el eje Y porque hay muchos más genes (1784) que disminuyen su expresión.

plotSmear contraste ELIvsSFI

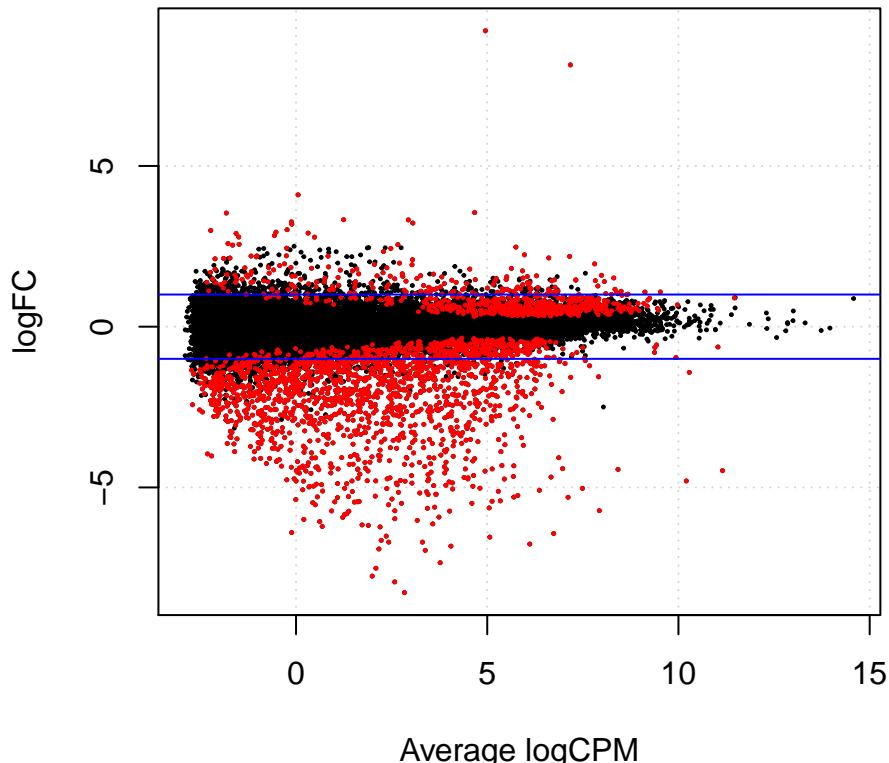


Figure 15: PlotSmear para el contraste ELI vs SFI.

Una vez tenemos la lista de genes DE en cada contraste, podemos buscar los genes diferencialmente expresados comunes a dos contrastes distintos. Por ejemplo, los genes que comúnmente cambian su expresión entre los contrastes ELIvsNIT y SFIvsNIT.

```

## [1] 3

## [1] "16760" "6717" "3379"

```

Existen 3 genes DE comunes para los contrastes ELIvsNIT y SFIvsNIT. Estos genes corresponden a las filas: 16760, 6717 y 3379; extrayendo estas filas en el archivo *counts* obtuve el identificador ENSEMBL de dichos genes. Es posible buscar el símbolo de estos genes directamente en el objeto **geneSymbols** creado a partir de las filas del objeto *fit* y el paquete de anotaciones *org.Hs.eg.db*. Sin embargo, en este caso sólo existe anotación del símbolo para uno de los tres genes comunes entre los contrastes ELIvsNIT y SFIvsNIT.

```
## [1] "ENSG00000170476" "ENSG00000239951" "ENSG00000143297"
## [1] NA      "SRI"   NA
```

Para a la siguiente pareja de contrastes ELIvsNIT y ELIvsSFI se encontraron 7 genes comunes que cambiaron su expresión en ambos contrastes.

```
## [1] 7
## [1] "49043" "42101" "48230" "31449" "45681" "3388"  "49407"
```

Buscamos el identificador ENSEMBL para estos siete genes al igual que su símbolo, para los cuales sólo existe anotación en uno de ellos.

```
## [1] "ENSG00000105369" "ENSG00000177455" "ENSG00000167483" "ENSG00000156738"
## [5] "ENSG00000007312" "ENSG00000163534" "ENSG00000104894"
## [1] "ICR1" NA      NA      NA      NA      NA
```

La última pareja de contrastes a comparar fue SFIvsNIT con ELIvsSFI, para la cual no se encontraron genes DE comunes.

5.2 Búsqueda de patrones de expresión y agrupación de las muestras

Como ya se mencionó en el apartado “Métodos” he realizado tres contrastes distintos para llevar a cabo la expresión diferencial de los genes entre más de dos grupos. Estos tres contrastes se recogen en la *table 5* presentada en el apartado anterior.

El resultado de la búsqueda de patrones a través del método *ANOVA* es una tabla llamada **topTags_anov** donde se recogen los genes comunes a los tres contrastes que más han cambiado su expresión.

```
## Coefficient: LR test on 2 degrees of freedom
##          logFC.SFIvsNIT logFC.ELIvsNIT logFC.ELIvsSFI    logCPM      F
## 42101     1.40383287     8.220068    6.816235 4.051008 89.59959
## 48230     0.73921380     6.868999    6.129785 3.576271 84.41009
## 3388      0.44604124     7.789778    7.343736 3.769604 83.06057
## 49043     2.38653678     8.928706    6.542169 5.067141 82.72705
## 31449     0.73436065     7.496139    6.761778 6.112780 76.80327
## 53056     1.26275696     7.787057    6.524300 2.359229 72.27737
## 45681     0.51667085     5.065737    4.549066 3.887904 72.17215
## 45890     0.02717489     3.508652    3.481477 2.640516 69.42867
## 49407     0.36688810     4.253443    3.886555 5.358100 68.24929
## 4412      0.93413976     6.958859    6.024719 3.169351 68.05349
##          PValue        FDR
## 42101 1.609672e-13 2.609321e-09
```

```

## 48230 3.258795e-13 2.609321e-09
## 3388 4.309989e-13 2.609321e-09
## 49043 4.539924e-13 2.609321e-09
## 31449 1.178397e-12 5.418270e-09
## 53056 2.338512e-12 7.998554e-09
## 45681 2.435401e-12 7.998554e-09
## 45890 3.174650e-12 9.123149e-09
## 49407 4.927521e-12 1.174404e-08
## 4412 5.108325e-12 1.174404e-08

```

Además, he seleccionado el número total de genes diferencialmente expresados en cada grupo con un FDR del 5%. He obtenido un total de 2277 genes que cambian su expresión con respecto a los 22990 genes analizados, que era el número de genes seleccionados tras la filtración.

```

##           LR test on 2 degrees of freedom
## NotSig                         20713
## Sig                            2277

```

5.3 Análisis de significación biológica

Para el análisis de significación biológica se identificaron los términos GO que están sobrerepresentados en los tres grupos: ELI, SFI y NIT. Esto es posible aplicando la función **goana** a los resultados de expresión diferencial de estas comparaciones. Con la función **topGo** he obtenido una tabla con los 20 términos GO más enriquecidos en los tres grupos. Esta tabla está registrada en el directorio **results** como *topGO* y también se muestra a continuación.

	Term	Ont	N	DE
## GO:0004896	cytokine receptor activity	MF	46	13
## GO:0014731	spectrin-associated cytoskeleton	CC	5	4
## GO:2000209	regulation of anoikis	BP	12	6
## GO:0098531	ligand-activated transcription factor activity	MF	21	8
## GO:0004879	nuclear receptor activity	MF	21	8
## GO:0140375	immune receptor activity	MF	56	14
## GO:2000811	negative regulation of anoikis	BP	9	5
## GO:0071415	cellular response to purine-containing compound	BP	3	3
## GO:0071394	cellular response to testosterone stimulus	BP	3	3
## GO:0009755	hormone-mediated signaling pathway	BP	85	18
## GO:0032754	positive regulation of interleukin-5 production	BP	6	4
## GO:0008091	spectrin	CC	6	4
## GO:0071383	cellular response to steroid hormone stimulus	BP	93	19
## GO:0003707	steroid hormone receptor activity	MF	24	8
## GO:0032634	interleukin-5 production	BP	10	5
## GO:0032674	regulation of interleukin-5 production	BP	10	5
## GO:0043401	steroid hormone mediated signaling pathway	BP	67	15
## GO:0071346	cellular response to interferon-gamma	BP	61	14
## GO:0043276	anoikis	BP	15	6
## GO:0032870	cellular response to hormone stimulus	BP	261	40
##	P.DE			
## GO:0004896	0.0002751658			
## GO:0014731	0.0003955261			
## GO:2000209	0.0004353688			
## GO:0098531	0.0004638715			

```

## GO:0004879 0.0004638715
## GO:0140375 0.0006451524
## GO:2000811 0.0007425973
## GO:0071415 0.0008928711
## GO:0071394 0.0008928711
## GO:0009755 0.0010005902
## GO:0032754 0.0010959626
## GO:0008091 0.0010959626
## GO:0071383 0.0011566294
## GO:0003707 0.0012919130
## GO:0032634 0.0013679971
## GO:0032674 0.0013679971
## GO:0043401 0.0014362579
## GO:0071346 0.0015933781
## GO:0043276 0.0018317233
## GO:0032870 0.0018422185

```

Los nombres de las filas de la salida son los identificadores universales de los términos GO, con un término por fila. Para entender bien el resultado obtenido, haré una descripción de lo que representa cada columna de la tabla *topGo*:

Term: Nombra los términos GO
Ont: Conjunto de dominios o tipos de términos -BP: Proceso Biológico -CC: Componente Celular -MF: Función Molecular
N: Número total de genes que se anotan en cada término
GO DE: Número de genes correspondiente a dicho término GO que están diferencialmente expresados
P.DE: Contiene los valores p para la sobrerepresentación del término GO en el conjunto de genes

La tabla de salida está ordenada por orden ascendente del p-valor, es decir los términos más afectados están al inicio de la tabla, lo que significa que el proceso más afectado es la actividad del receptor de citoquinas.

5.3.1 Probando conjuntos de genes (Gene Set Testing)

Para profundizar un poco más en el análisis de enriquecimiento biológico, utilicé la prueba del conjunto de genes de rotación: **ROAST**. Dado un conjunto de genes, es posible probar si la mayoría de los genes del conjunto están diferencialmente expresados a través del contraste o contrastes de interés. Es útil cuando el conjunto especificado contiene todos los genes involucrados en alguna vía o proceso.

Para este caso concreto, me interesé en dos términos GO relacionados con la actividad del receptor _GO:0004896_ y _GO:0004879_; puesto que en la tabla anterior existen dos términos relacionados con esta función molecular. Ambos términos se han usado para definir un conjunto que contenga todos los genes que están anotados con cada término.

El siguiente paso fue ejecutar el test **ROAST** en los conjuntos de genes definidos para el contraste de interés. Supongamos que la comparación de interés es entre los grupos ELI (Extensive lymphoid infiltrates) y NIT (not infiltrated tissues). Utilicé la función **fry** para probar múltiples conjuntos de genes, obteniendo como resultado la siguiente tabla.

	NGenes	Direction	PValue	FDR	PValue.Mixed	FDR.Mixed
## GO:0004896	46	Up	0.0002531509	0.0005063017	5.679592e-10	1.135918e-09
## GO:0004879	21	Up	0.0267944576	0.0267944576	2.415503e-05	2.415503e-05

Cada fila del objeto *fr* (que representa la tabla anterior) corresponde a un único conjunto de genes; en este caso de cada término GO. Explicaré cada columna de la tabla *fr* para entenderla mejor:

NGenes: Número de genes en cada conjunto de genes
Direction: Es la dirección neta del cambio, esta se determina a partir de la importancia de los cambios en cada dirección
PValue: Sirve para determinar

si la mayoría de los genes del conjunto se expresan diferencialmente en la dirección especificada **FDR**: Este estadístico se calcula a partir de los correspondientes p-valores en todos los conjuntos **PValue.Mixed**: Prueba la expresión diferencial en cualquier dirección

Para finalizar, he representado visualmente el resultado anterior a través de la función **barcodeplot**. Esta función permite visualizar los resultados de cualquier conjunto presente en la tabla *fr*; en este caso, he escogido el conjunto de genes definido por [_GO:0004896](#).

Los genes están representados por barras y se clasifican de izquierda a derecha disminuyendo el cambio de pliegue, formando así el patrón de código de barras. La línea sobre el código de barras muestra el enriquecimiento local relativo de las barras verticales en cada parte del gráfico. Este gráfico en particular sugiere que la mayoría de los genes de este conjunto están down-regulated en el grupo ELI en comparación con el grupo NIT.

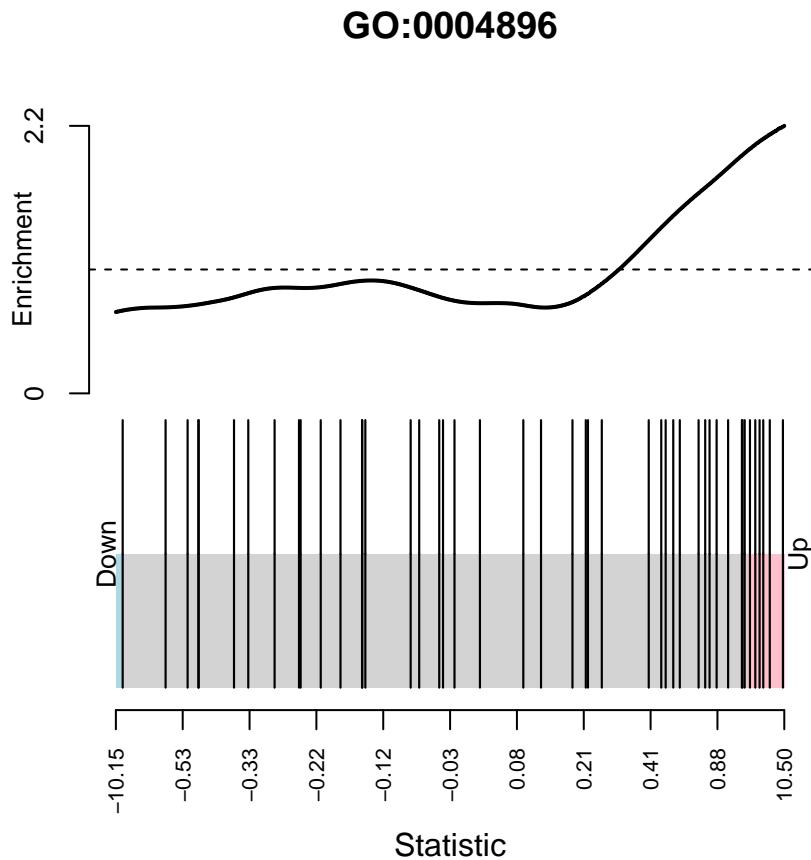


Figure 16: BarcodePlot ELI vs NIT.

6 Discusión

Para la realización de este análisis he encontrado varias dificultades, entre ellas la falta de contexto o información del experimento del cual se han obtenido los datos. Toda la información que se ha dado junto a los datos ha sido el portal donde se recogen los datos y lo que significa cada grupo, pero no se ha dado más información cerca del objetivo del experimento, ni de las preguntas biológicas que se quieren responder.

Otra gran dificultad es la gran cantidad de valores para los que no existe anotación, obteniendo tablas con una gran cantidad de valores perdidos o NA, por lo que esto ha dificultado aún más la posible interpretación de los resultados.

Por último, existen varios paquetes para el análisis de datos de RNA-seq, entre ellos destacan: DESeq, DESeq2, limma o el paquete edgeR que ha sido el que yo he elegido. La gran diferencia entre estos paquetes es el modelo probabilístico que utilizan para calcular la expresión diferencial de los genes. He escogido el paquete edgeR porque creo que es un paquete fácil de utilizar y porque su tiempo de cómputo para ejecutar los comando es medio, pero también presenta limitaciones como su alta permisividad a la hora de considerar un gen diferencialmente expresado o no. (5)

7 Conclusión

Debido a la falta de contexto e información adicional las conclusiones que puedo sacar es que los procesos biológicos más afectados son los relacionados con la actividad de los receptores y que numerosas vías de señalización de hormonas se ven implicados en dichos cambios también.

8 Setup

```
## R version 4.0.1 (2020-06-06)
## Platform: x86_64-apple-darwin17.0 (64-bit)
## Running under: macOS Catalina 10.15.4
##
## Matrix products: default
## BLAS:    /Library/Frameworks/R.framework/Versions/4.0/Resources/lib/libRblas.dylib
## LAPACK:  /Library/Frameworks/R.framework/Versions/4.0/Resources/lib/libRlapack.dylib
##
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## attached base packages:
## [1] parallel  stats4    stats     graphics  grDevices utils      datasets
## [8] methods   base
##
## other attached packages:
##  [1] GenomicRanges_1.40.0 GenomeInfoDb_1.24.0  GO.db_3.11.4
##  [4] RColorBrewer_1.1-2   org.Hs.eg.db_3.11.4  AnnotationDbi_1.50.0
##  [7] IRanges_2.22.2       S4Vectors_0.26.1   Biobase_2.48.0
## [10] BiocGenerics_0.34.0 dplyr_1.0.0       edgeR_3.30.3
## [13] limma_3.44.3        readr_1.3.1       float_0.2-4
##
## loaded via a namespace (and not attached):
##  [1] Rcpp_1.0.4.6          locfit_1.5-9.4      lattice_0.20-41
##  [4] prettyunits_1.1.1     Biostrings_2.56.0    assertthat_0.2.1
##  [7] digest_0.6.25         BiocFileCache_1.12.0 R6_2.4.1
## [10] RSQLite_2.2.0         evaluate_0.14       httr_1.4.1
## [13] highr_0.8            pillar_1.4.4        zlibbioc_1.34.0
## [16] rlang_0.4.6           progress_1.2.2      curl_4.3
## [19] blob_1.2.1            rmarkdown_2.2        splines_4.0.1
## [22] statmod_1.4.34       stringr_1.4.0       RCurl_1.98-1.2
## [25] bit_1.1-15.2          biomaRt_2.44.0      compiler_4.0.1
## [28] xfun_0.14             pkgconfig_2.0.3     askpass_1.1
## [31] htmltools_0.4.0        openssl_1.4.1       tidyselect_1.1.0
## [34] tibble_3.0.1           GenomeInfoDbData_1.2.3 XML_3.99-0.3
## [37] dbplyr_1.4.4           crayon_1.3.4        rappdirs_0.3.1
```

```

## [40] bitops_1.0-6           grid_4.0.1          lifecycle_0.2.0
## [43] DBI_1.1.0              magrittr_1.5        stringi_1.4.6
## [46] XVector_0.28.0         ellipsis_0.3.1    generics_0.0.2
## [49] vctrs_0.3.1            tools_4.0.1         bit64_0.9-7
## [52] glue_1.4.1              purrr_0.3.4        hms_0.5.3
## [55] yaml_2.2.1             BiocManager_1.30.10 memoise_1.1.0
## [58] knitr_1.28

```

Bibliografía

1. Lonsdale J, Thomas J, Salvatore M, Phillips R, Lo E, Shad S, et al. The genotype-tissue expression (gtex) project. *Nature genetics*. 2013;45(6):580.
2. Robinson MD, McCarthy DJ, Smyth GK. EdgeR: A bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics*. 2010;26(1):139–40.
3. Rohart F, Gautier B, Singh A, Lê Cao K-A. MixOmics: An r package for ‘omics feature selection and multiple data integration. *PLoS computational biology*. 2017;13(11):e1005752.
4. Robinson MD, Oshlack A. A scaling normalization method for differential expression analysis of rna-seq data. *Genome biology*. 2010;11(3):R25.
5. Kvam VM, Liu P, Si Y. A comparison of statistical methods for detecting differentially expressed genes from rna-seq data. *American journal of botany*. 2012;99(2):248–56.