PEC2

Beatriz Pardo Montenegro

7/6/2020

Table of Contents

URL GitHub: https://github.com/bpardom/AO_PEC2.git

1.Abstract

Mediante análisis de RNA seq comparamos la expresión génica en distintos tipos de muestras de tiroides, según presenten algún grado o no de infiltración linfoide. Lo genes que están diferencialmente expresados en cada uno de los 3 grupos vemos que están implicados en procesos biológicos de respuesta inmune (GO:0006955, GO:0002376) y de adhesión biológica (GO:0022610).

2.Objetivos

Evaluar el expresión génica diferencial entre en muestras con distinto grado de infiltración linfoide. Queremos saber también en qué procesos biológicos y rutas metabólicos están implicados los genes diferencialmente expresados según el grado de infiltración.

3.Materiales y métodos

3.1 Naturaleza de los datos

Parto de los datos de expresión (RNA-seq) pertenecientes a un análisis del tiroides, donde se compara tres tipos de tejido que se diferencian según el grado de infiltración linfoide. Parto de un total de 292 muestras pertenecientes a tres grupos: • Not infiltrated tissues (NIT): 236 muestras • Small focal infiltrates (SFI): 42 muestras • Extensive lymphoid infiltrates (ELI): 14 muestras

Para mi análisis debo tomar al azar 10 muestras de cada grupo de los 3 que hay en el archivo targets y una vez seleccionadas debo conseguir que el programa cargue los datos de expresión de dichas muestras del archivo counts.

3.2 Tipo de experimento

El tipo de experimento que planteo es de comparación de grupos. El objetivo de los estudios comparativos es determinar si los perfiles de expresión génica difieren entre grupos previamente identificados, en mi análisis tengo 10 muestras de tejido sin infiltración linfoide, 10 muestras con pequeños focos de infiltración y 10 muestras con una extensa infiltración linfoide.

3.3 Materiales

3.3.1 Datos

Utilizo los datos de 2 archivos targets y counts que contienen la información de las muestras de un estudio obtenido del repositorio (GTEx1). Este repositorio contiene datos de múltiples tipos en un total de 54 tejidos. Nosotros nos centraremos en los datos de expresión (RNA-seq) pertenecientes a un análisis del tiroides en donde se compara tres tipos de infiltración medido en un total de 292 muestras pertenecientes a tres grupos. Seleccionamos 10 muestras de cada uno de los grupos.

3.3.2 Sofware utilizado

Para el desarrollo del proceso de análisis de los datos, utilizo el software libre R a través de la interfaz RStudio. Los paquetes que fueron usados para la realización del proyecto, provienen tanto de R como de Bioconductor. R es un lenguaje de programación funcional orientado especialmente a la manipulación de datos, cálculos

estadísticos y generación y visualización de gráficos. Por su parte, RStudio es un entorno de desarrollo integrado y Bioconductor es un software libre que utiliza el lenguaje estadístico de R y proporciona herramientas para el análisis y comprensión de datos genómicos de alto rendimiento. Para que el experimento sea reproducible y poderlo compartir con otras personas creo un repositorio en Github.

3.4 Procedimiento general de análisis

3.4.1 Definición de los datos

Cargo los datos del archivo targets y del archivo counts mediante la función read.csv. Fijo una semilla, doy la orden para que me coja aleatoriamente 10 muestras de cada grupo y que no haya reemplazo, para que siempre sea la misma muestra. Corrijo el posible conflicto entre los . y los - entre el archivo targets y counts.Selecciono que se carguen las 10 filas de cada grupo de targets cogiendo las columnas del archivo de counts cuyo nombre de la muestra coincide. Nombro a cada muestra del grupo ELI como ELI1, ELI2, ELI3,… y de igual manera con NIT y SFI.

Compruebo que son 56202 genes.

3.4.2 Preprocesado de los datos: filtraje y normalización

Creo un objeto DGElist a partir de la matriz de conteos. Filtro los genes que tengan una baja expresión en la mayoria de muestras. Para cuantificar la expresión utilizamos la función cpm que calcula los valores de recuento por millón. Se filtran los genes que no superan un umbral de al menos dos muestras con más de un recuento por millón.

La cantidad de genes tras el filtrado se reduce considerablemente: 19537

Hago una transformación logarítmica de los datos mediante la funcion rlog para poder visualizar los datos. Y realizo control de calidad.

Creo un Heatmap para poder observar de una manera más visual la distancia entre muestras. Los colores no muestran una escala real de asociación pero ayudan al usuario a ver cómo se relacionan las muestras.

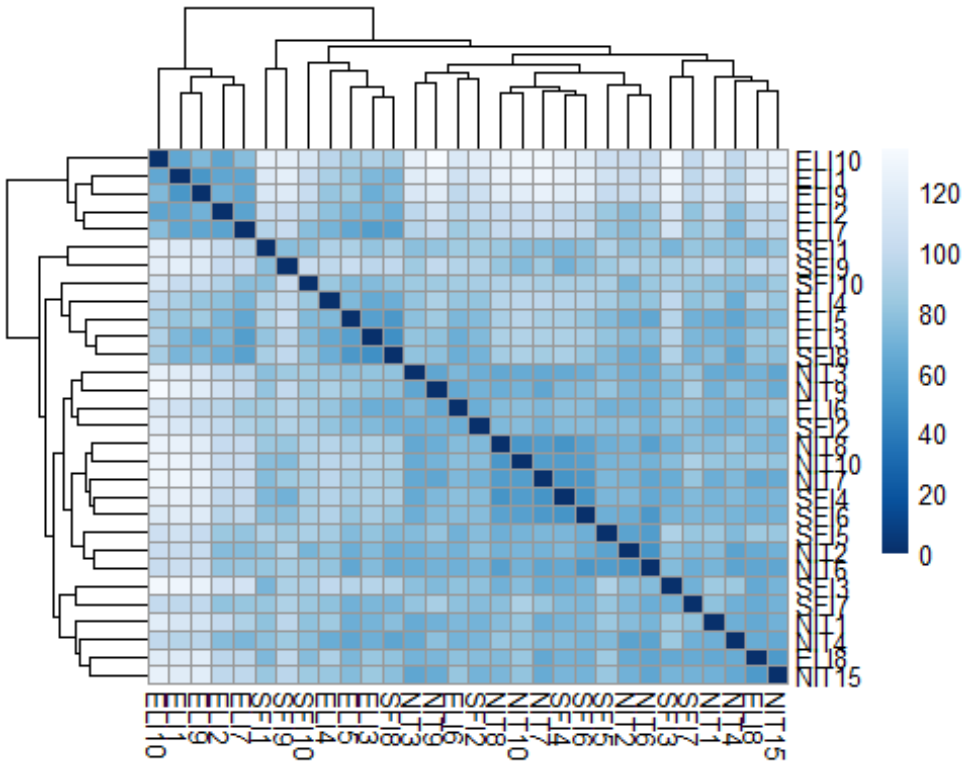


Figura1: Heatmap representativo de las distancias entre las muestras

Instalo la librería pcaExplorer que contiene funciones que permiten hacer gráficos muy visuales. En los gráficos de análisis de componentes principales se observan como se agrupan las muestras. El primer componente explica un 59,97% de la variabilidad de la varianza.

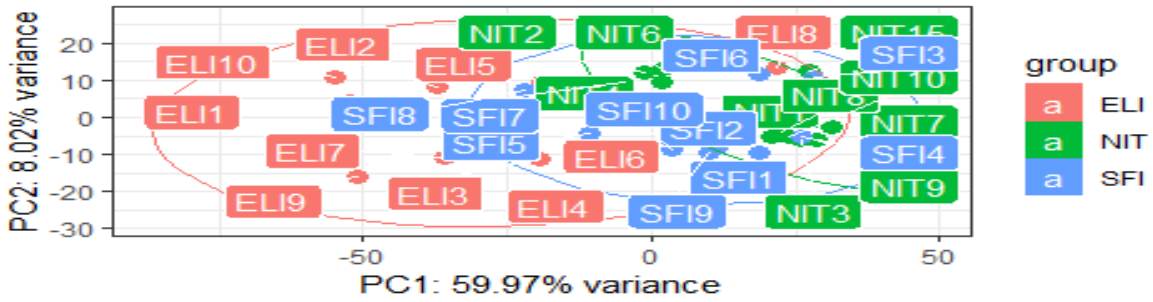


Figura2: Plot análisis componentes principales

La figura 3 muestra la distribución del recuento de cada muestra después de filtrar 36665 genes que mostraron baja expresión. Continúo trabajando con la expresión de 19537 genes.

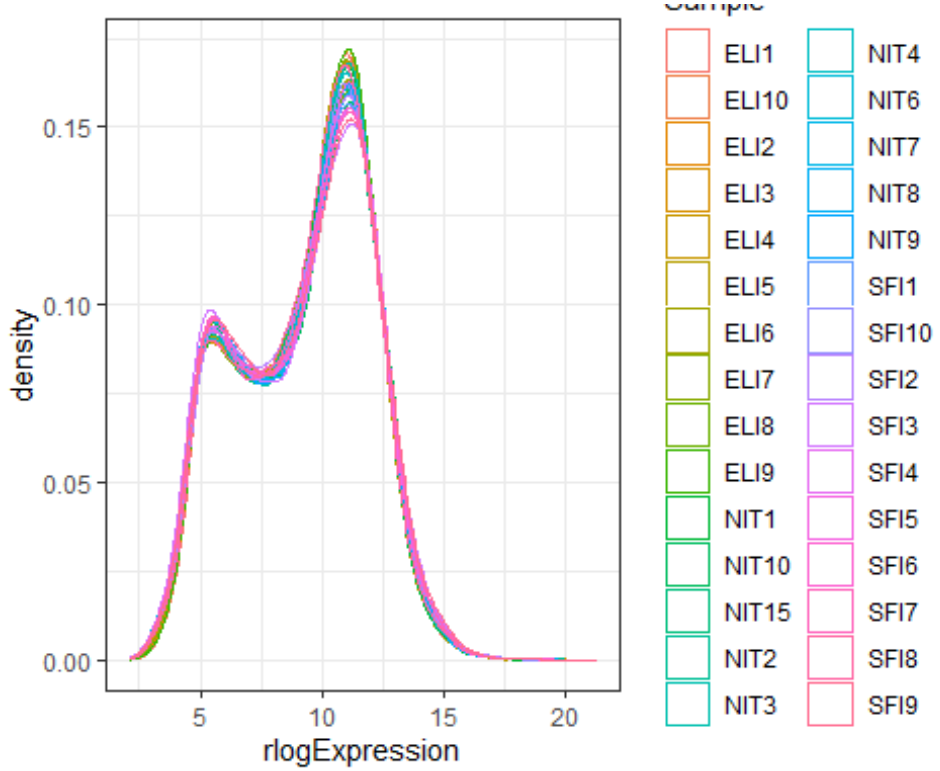


Figura3: Distribución del recuento de cada muestra

Cómo se aprecia en las figuras de boxplots todas las cajas son similares, no observándose cajas notablemente desplazadas hacia arriba o hacia abajo, con lo que no debería descartar ninguna muestra.

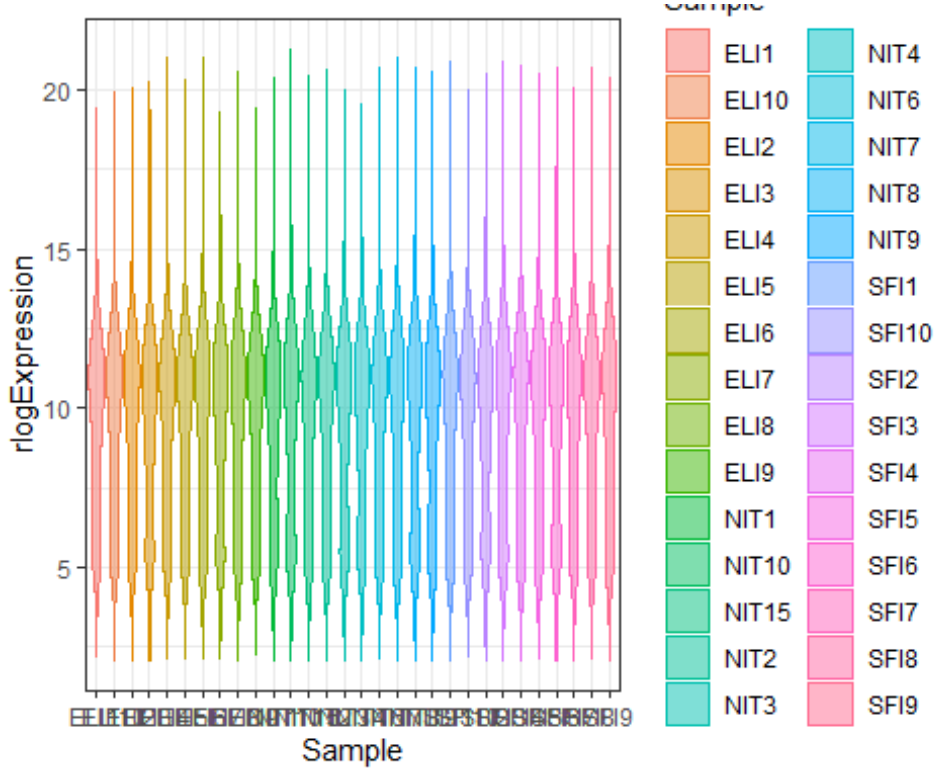


Figura4: Gráfico de distribución

Los genes con una alta (anormalmente grande) expresión corresponden con el bigote superior en los diagramas de caja que se representan en la figura 6.

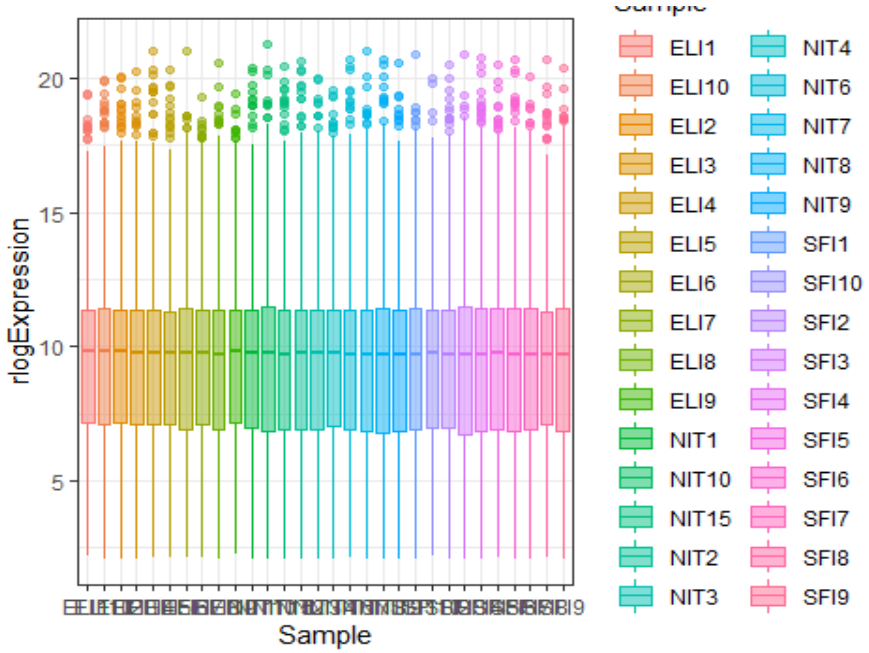


Figura5: Boxplot

La normalización es un proceso diseñado para identificar y eliminar las diferencias técnicas entre las muestras. Muchas son las ventajas de este paso del análisis, las dos principales: nos permite eliminar el ruido de fondo y nos permite hacer comparables todos los valores del estudio. Creo la matriz de diseño y normalizo con la función calcNormFactors. Utilizo la función voom para realizar una transformación en la que se estima la tendencia de la varianza respecto a la media en el counting data, ajustando la heterocedasticidad.

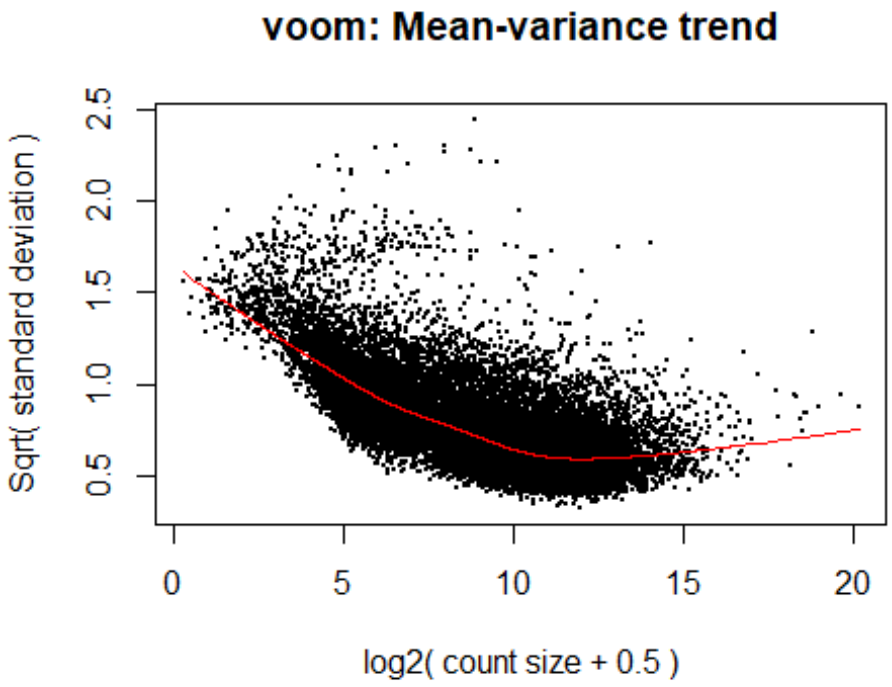


Figura6: Gráfico tranformación Voom tras normalización

3.4.3 Identificación de genes diferencialmente expresados

Realizo el análisis de expresión diferencial con DESeq2[1]. Es una manera muy cómoda ya que no es necesario realizar los diferentes pasos del análisis uno a uno. La mayoría de las funciones se han unificado y a través de la función DESeq y la función results para visualizar los resultados realizo el análisis completo en cada uno de los grupos.

La tabla de resultados proporcionada por la función results contiene información del pvalor y del pvalor ajustado (padj), a partir de los cuales obtenemos los tránsitos diferencialmente expresados.

[1] https://bioconductor.org/packages/release/bioc/html/DESeq2.html

Para conocer el número de genes diferencialmente expresados entre los 2 grupos experimentales utilizo la función summary. El paquete DESeq2 emplea por defecto el nivel de significación de alpha 0.01. Puedo modificarlo pero para poder comparar resultados el nivel de significación tiene que ser el mismo para todos los grupos.

En resc1, 1725 genes presentan un logFC significativamente negativo y 2530 genes un logFC significativamente positivo a un nivel de significación de 0.01.

En resc2, 1207 genes presentan un logFC significativamente negativo y 2603 genes un logFC significativamente positivo a un nivel de significación de 0.01.

En resc3, 70 genes presentan un logFC significativamente negativo y 193 genes un logFC significativamente positivo a un nivel de significación de 0.01.

Si bajo la significación a 0.05 se observa un descenso en los genes diferencialmente expresados tanto positivos como negativos. En resc1 1071 genes presentan un logFC significativamente negativo y 1943 genes un logFC significativamente positivo a un nivel de significación de 0.05.En resc2 653 genes presentan un logFC significativamente negativo y 2139 genes un logFC significativamente positivo a un nivel de significación de 0.05. En resc3 17 genes presentan un logFC significativamente negativo y 81 genes un logFC significativamente positivo a un nivel de significación de 0.05.

Los resultados del análisis de expresión diferencial con DESeq2 los visualizo mediante un gráfico MAplot.Este gráfico representa lea media de lecturas normalizadas de cada gen frente al logaritmo de base 2 del fold change.

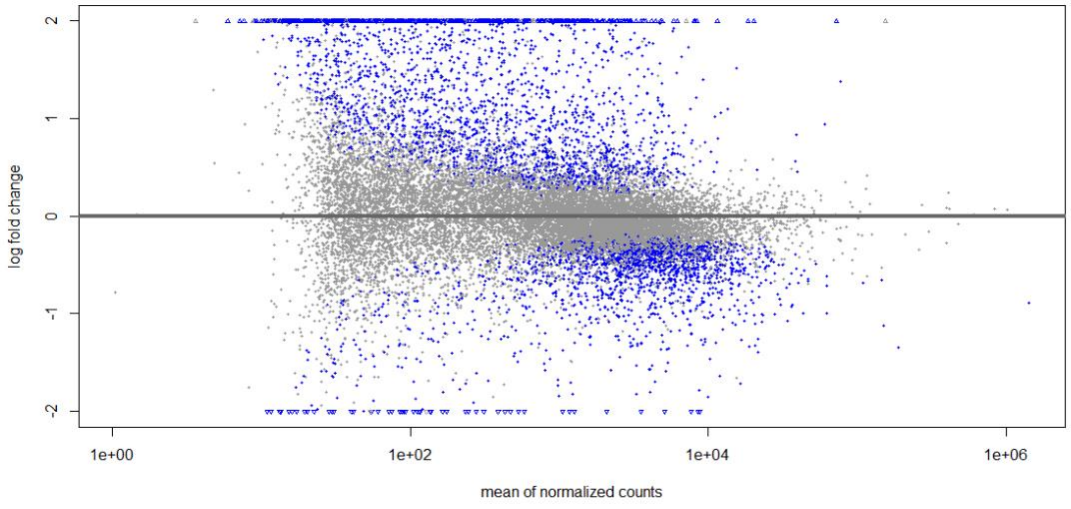


Figura7: Gráfico MAplot resc1 expresión diferencial

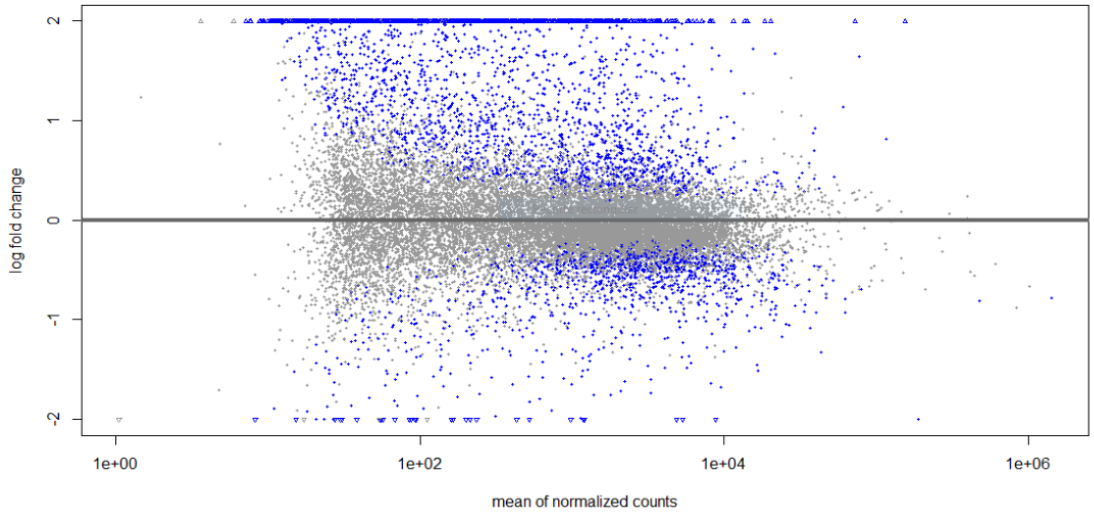


Figura8: Gráfico MAplot resc2 expresión diferencial

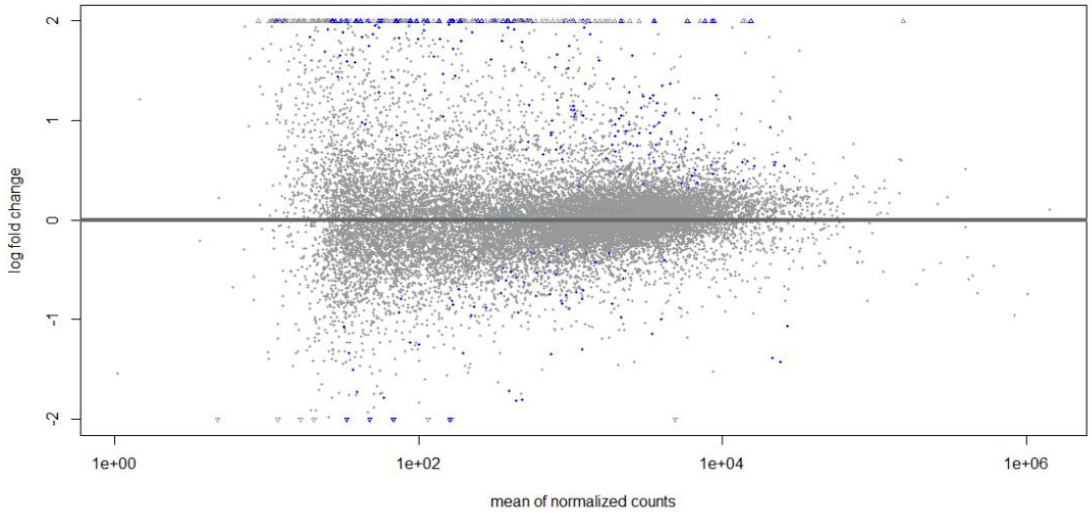


Figura9: Gráfico MAplot resc3 expresión diferencial

El paquete DESeq2 también incluye una función con la que reducir el efecto del tamaño siendo útil para la visualización y obtener un ranking adecuado de los genes. La reducción LFC se realiza con la función lfcShrink. Lo realizo para resc1, selecciono la reducción apeglm y represento graficamente.

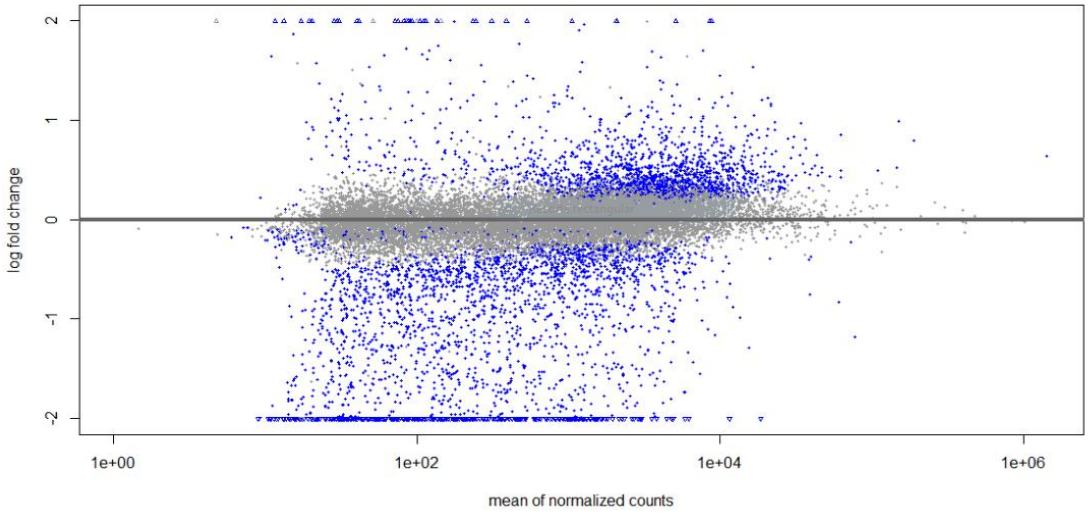


Figura10: Gráfico MAplot resc1 con reducción LFC

3.4.4 Anotación de los resultados

Realizo la anotación de los genes con la función mapIds. Repito la operación para cada grupo.

3.4.5 Comparación entre las distintas comparaciones

Partíamos de tres grupos de muestras, 10 muestras de ELI, 10 muestras de NIT y 10 muestras de SFI. Vamos a realizar la comparación entre las distintas comparaciones. En el 3.4.3 calculé los genes diferencialmente expresados con la función DESeq y results del paquete DESeq2 comparando los grupos ELI, NIT y SFI 2 a 2. Ahora lo voy a hacer con la función lmFit de la librería limma.

Creo la matriz de contraste.

```
##       Contrasts
## Levels ELI - NIT ELI - SFI NIT - SFI
##    ELI         1         1         0
##    NIT        -1         0         1
##    SFI         0        -1        -1

##        ELI - NIT ELI - SFI NIT - SFI
## Down         343       507         0
## NotSig     17842     17691     19537
## Up          1352      1339         0
```

Los resultados realizando la comparación de comparaciones con el modelo lineal de la librería limma da unos resultados de genes diferencialmente expresados todavía más bajos que los realizados con DESeq y results utilizando una significación de 0.05. De hecho para la comparación entre el grupo NIT y SFI no hay genes significativamente positivos ni negativos.

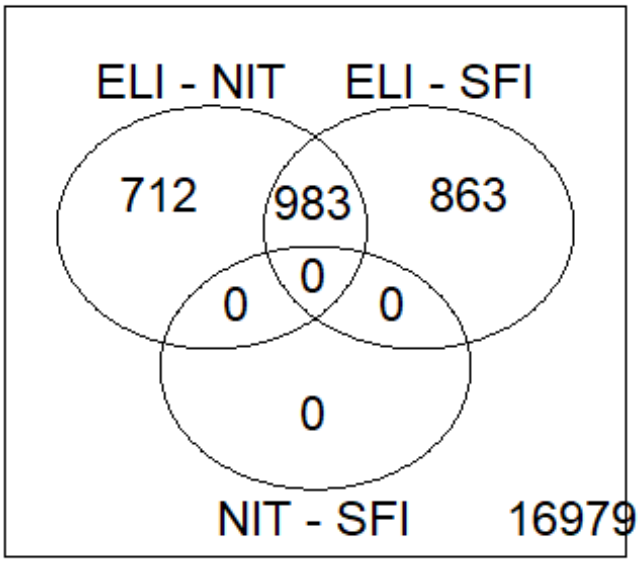


Figura11: Diagrama comparación comparaciones con libreria limma

3.4.6 Análisis de significación biológica

Realizo el análisis de significación biológica con la función goseq en cada uno de los grupos para ver en qué procesos biológicos y vías metabólicas se ven implicados los genes seleccionados. Con la función GOseq obtengo los términos de Gene Ontology de los genes diferencialmente expresados.

Términos Gene Ontology resc1: "GO:0006955" "GO:0002376" "GO:0050896" "GO:0023052" "GO:0007154" "GO:0007165"

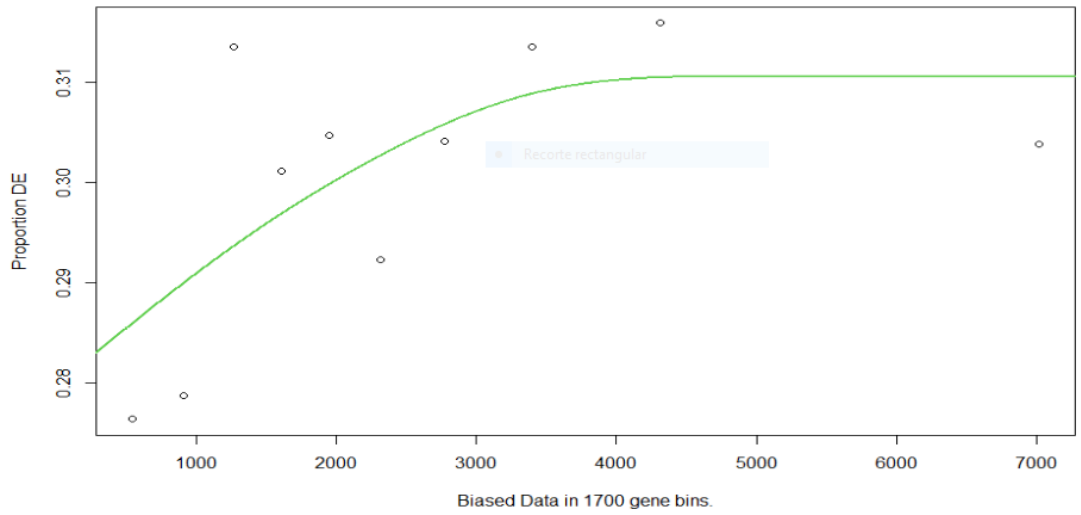


Figura12: Bondad de ajuste resc1

	category <chr>	over_represented_pvalue <dbl>	under_represented_pvalue <dbl>	numDEInCat <int>	numInCat <int>
3475	GO:0006955	5.484520e-39	1	740	1660
1004	GO:0002376	2.370685e-38	1	1028	2478
13419	GO:0050896	7.833587e-36	1	2523	7143
7022	GO:0023052	4.564483e-34	1	1885	5119
3572	GO:0007154	1.413214e-33	1	1888	5137
3583	GO:0007165	2.244684e-32	1	1737	4686

Figura13: Resultado función goseq

Términos Gene Ontology resc2:"GO:0002376" "GO:0006955" "GO:0002250" "GO:0046649" "GO:0002682" "GO:0045321"

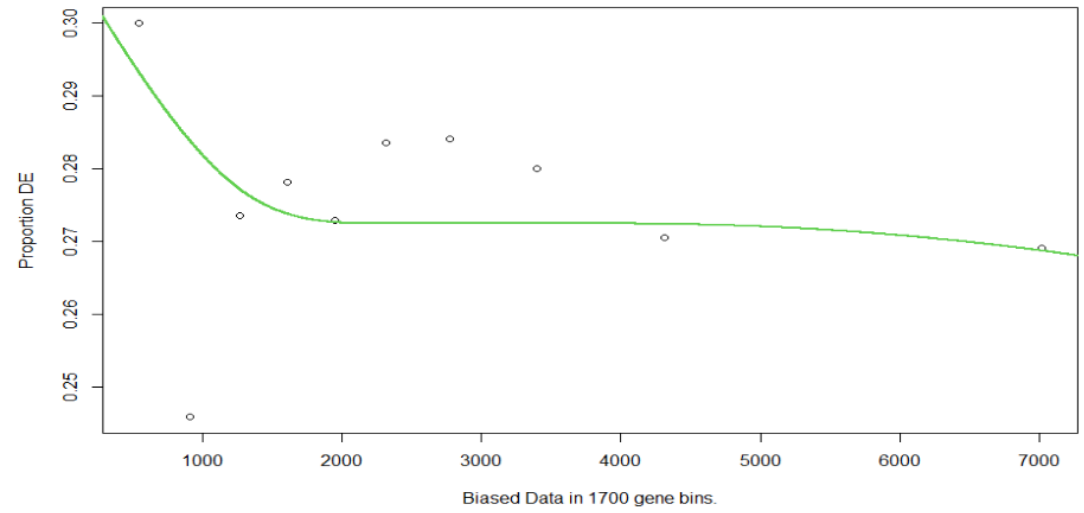


Figura14: Bondad de ajuste resc2

	category	over_represented_pvalue	under_represented_pvalue	numDEInCat	numInCat
	<chr>	<dbl>	<dbl>	<int>	<int>
1004	GO:0002376	3.969272e-53	1	1000	2478
3475	GO:0006955	4.057530e-53	1	729	1660
918	GO:0002250	2.925859e-41	1	221	364
12437	GO:0046649	2.117268e-40	1	313	594
1174	GO:0002682	4.326224e-40	1	555	1262
11821	GO:0045321	1.063764e-38	1	494	1097

Figura15: Resultado función goseq resc2

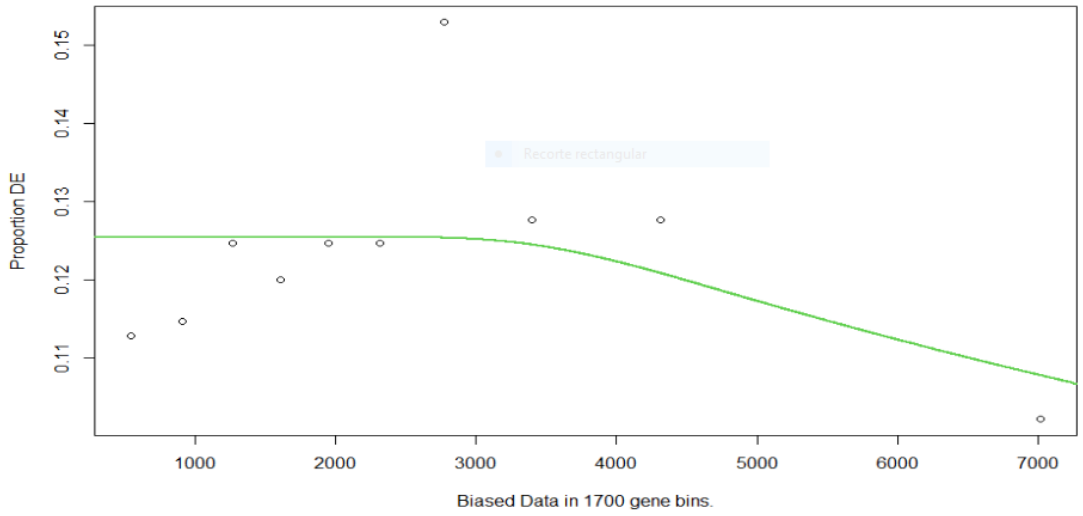


Figura16: Bondad de ajuste resc3

	category	over_represented_pvalue	under_represented_pvalue	numDEInCat	numInCat
	<chr>	<dbl>	<dbl>	<int>	<int>
6968	GO:0022610	2.508950e-13	1	238	1245
3573	GO:0007155	4.668833e-13	1	236	1239
7115	GO:0030155	1.798133e-10	1	130	616
918	GO:0002250	7.453122e-10	1	87	364
17755	GO:0098609	2.677545e-09	1	144	732
3577	GO:0007159	7.131923e-09	1	73	300

Términos Gene Ontology resc3: "GO:0022610" "GO:0007155" "GO:0030155" "GO:0002250" "GO:0098609" "GO:0007159"

4.Resultados

Utilizando DESeq2 con un nivel de significación de 0.05 evalúo la expresión diferencial comparando entre grupos resc1 (ELI-NIT): 1071 genes presentan un logFC significativamente negativo y 1943 genes un logFC significativamente positivo.En resc2 (ELI-SFI) 653 genes presentan un logFC significativamente negativo y 2139 genes un logFC significativamente positivo. En resc3 (NIT-SFI) 17 genes presentan un logFC significativamente negativo y 81 genes un logFC significativamente positivo.

5.Discusión

La tencnología RNA-seq está suponiendo una revolución de los estudios de transcriptómica pero todavía no se ha decidido la metodología estándar a seguir para el ánalisis de los datos especialmente los de expresión diferencial. En este trabajo hago la expresión diferencial con DESeq2 y con Limma, que son 2 de los más populares y los resultados presentan diferencias.

6.Apéndice

Tal y cómo se ha hablado en repetidas ocasiones, la reproducibilidad del estudio es fundamental a la hora de trabajar como bioinformáticos, con lo que creo un repositorio en Github con todo lo relativo al proyecto de forma que se pueda clonar en otro ordenador y reproducir mi trabajo[2].

URL(puesta también al inicio del informe): https://github.com/bpardom/AO_PEC2.git

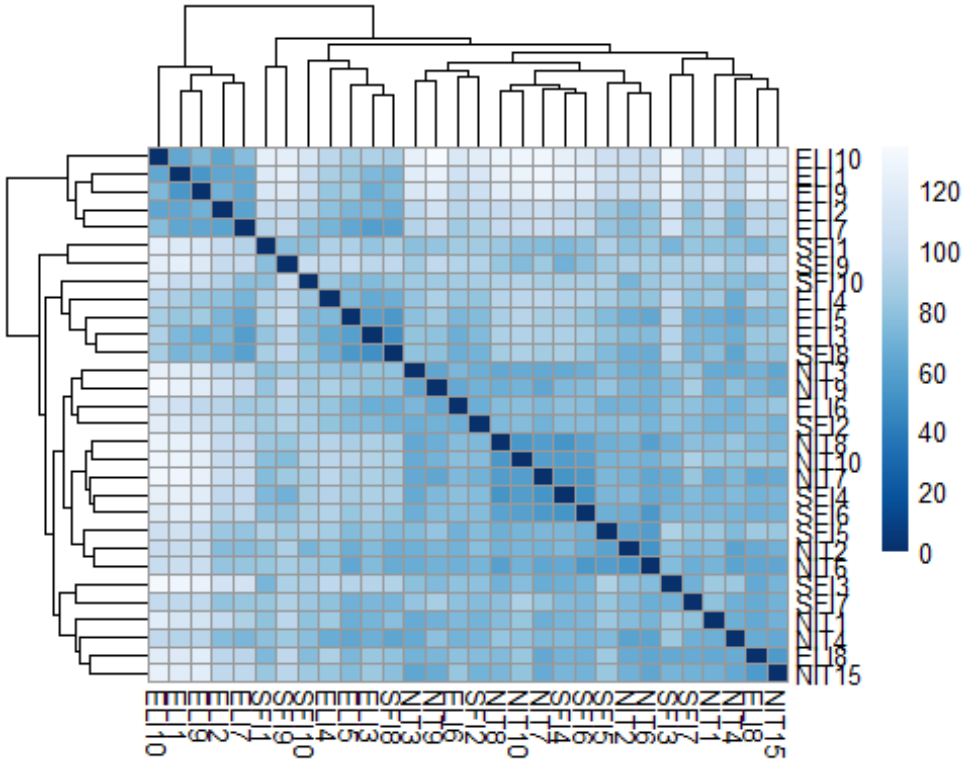
Pongo a continuación el código de R utilizado para la realización del análisis. También está disponible en el documento RMD disponible en repositorio GitHub.

```
#Cargo las librerías que utilizo para la realización del análisis
library(dplyr)
library(stringr)
library(BiocManager)
library(DESeq2)
library(edgeR)
library(DEFormats)
library(pheatmap)
library(RColorBrewer)
library(pcaExplorer)
library(apeglm)
library(limma)
library(ggbeeswarm)
library(AnnotationDbi)
library(org.Hs.eg.db)
library(grex)
library(clusterProfiler)
library(goseq)

#Cargo los datos utilizando read.csv
targets <- read.csv(file = "C:/Bea/Master/Datos omicos
Bea/PEC2/Data/targets.csv", header = T, sep = ",")
counts <- read.csv(file = "C:/Bea/Master/Datos omicos
Bea/PEC2/Data/counts.csv", header = T, sep = ";")
```

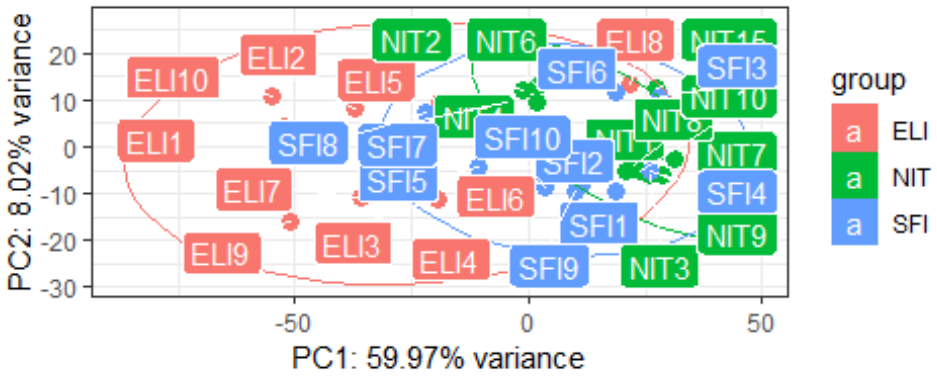
[2] https://cfss.uchicago.edu/setup/git-with-rstudio/

```r
#Fijo semilla y selecciono del archivo targets 10 de cada grupo sin
reemplazo
set.seed(123456)
selec<- targets %>% group_by(Group) %>% sample_n(size = 10, replace =
FALSE)

#Soluciono error entre . y - Cojo para cada muestra seleccionada en
targets la columna de counts cuyo nombre coincida
colnames(counts) <- str_replace_all(colnames(counts), "[.]", "-")
selecname <- c(selec$Sample_Name)
seleccount <- counts[2:293][selecname]
rownames(seleccount) <- counts[,1]

#Nombro cada columna del grupo ELI con números correlativos e igual para
NIT y SFI
cols<-
c("ELI1","ELI2","ELI3","ELI4","ELI5","ELI6","ELI7","ELI8","ELI9","ELI10",
"NIT1","NIT2","NIT3","NIT4","NIT15","NIT6","NIT7","NIT8","NIT9","NIT10","
SFI1","SFI2","SFI3","SFI4","SFI5","SFI6","SFI7","SFI8","SFI9","SFI10")
colnames(seleccount)<-cols

#Cantidad de genes
nrow(seleccount)

## [1] 56202

#Creación de un objeto DGElist a partir de la matriz de conteos
grupos <- rep(c("ELI", "NIT", "SFI"), each = 10)
seleclist <- DGEList(as.matrix(seleccount), group = grupos)

#Calculo de recuento por millón con cpm
selecDESmillion <- cpm(seleclist)
scmillion <- selecDESmillion > 1
scmillionk <- which(rowSums(scmillion) >= 2)
millionlist <- seleclist[scmillionk,]

#Cantidad de genes tras filtrado
nrow(millionlist)

## [1] 19537

#Transformación logarítmica con rlog
millionlist <- as.DESeqDataSet(millionlist)
millionlog <- rlog(millionlist, blind = FALSE)

## rlog() may take a few minutes with 30 or more samples,
## vst() is a much faster transformation

#Cálculo de las distancias entre muestras y su representación mediane
Heatmap
sdist <- dist(t(assay(millionlog)))
sampleDistMatrix <- as.matrix(sdist)
```

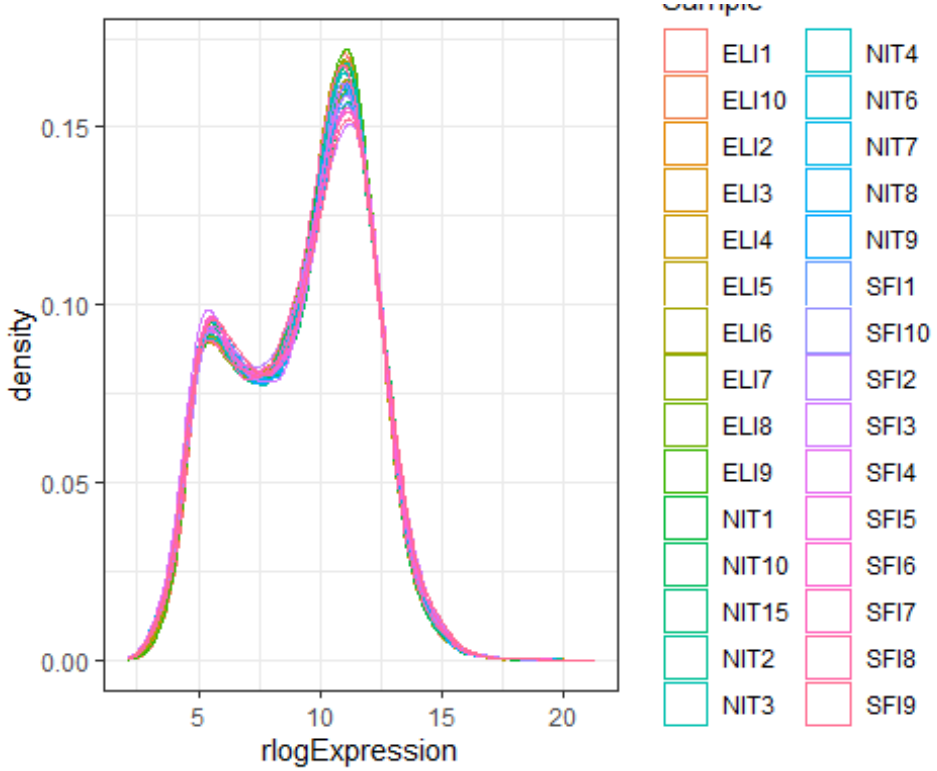
```
colors <- colorRampPalette( rev(brewer.pal(9, "Blues")) )(255)
pheatmap(sampleDistMatrix,
         clustering_distance_rows = sdist,
         clustering_distance_cols = sdist,
         col = colors)
```



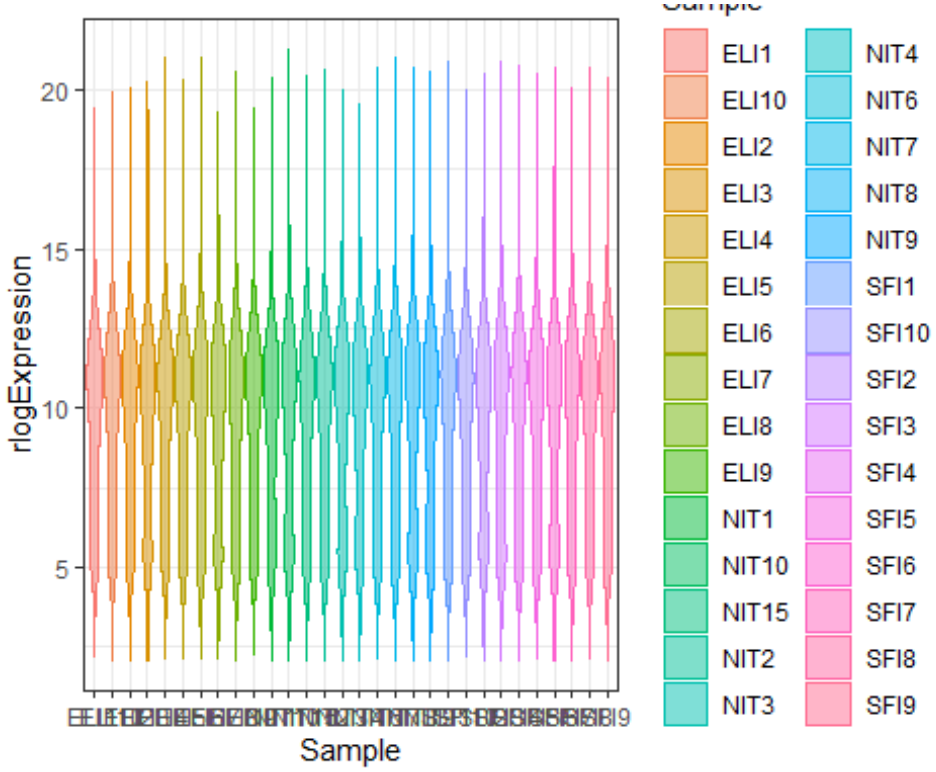

```
#Gráfico ánalisis de componentes
pcaplot(millionlog,intgroup = c("group"))
```



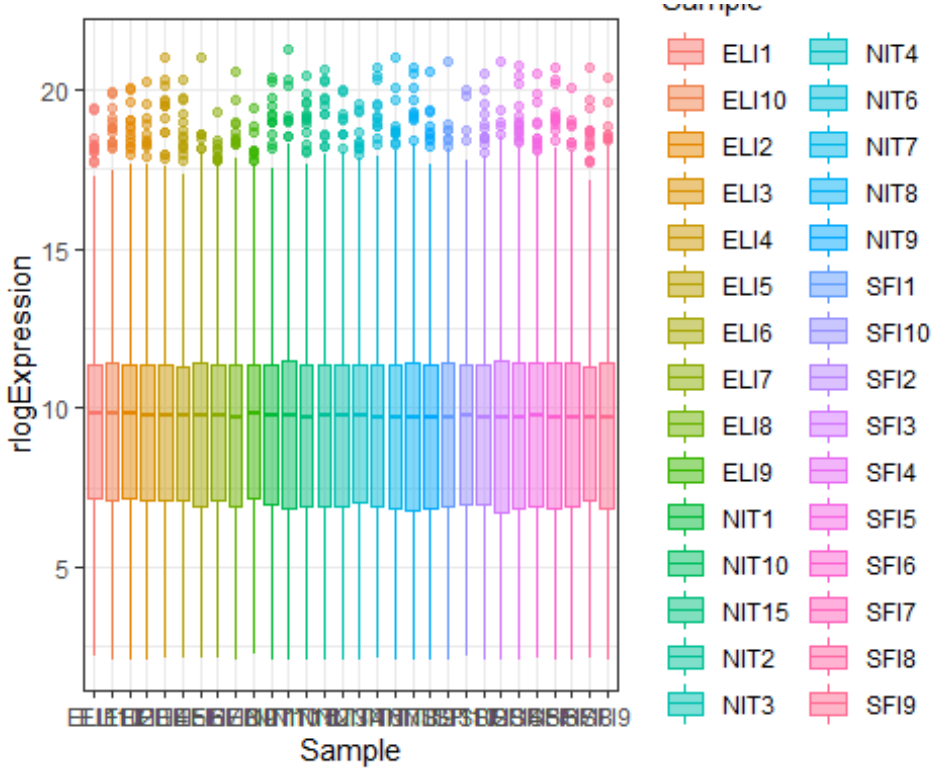

```
#Distribución del recuento de cada muestra
distro_expr(millionlog,plot_type = "density")
```



```
#Distribución de cada muestra
distro_expr(millionlog,plot_type = "violin")
```




```
#Boxplot
distro_expr(millionlog,plot_type = "boxplot")
```

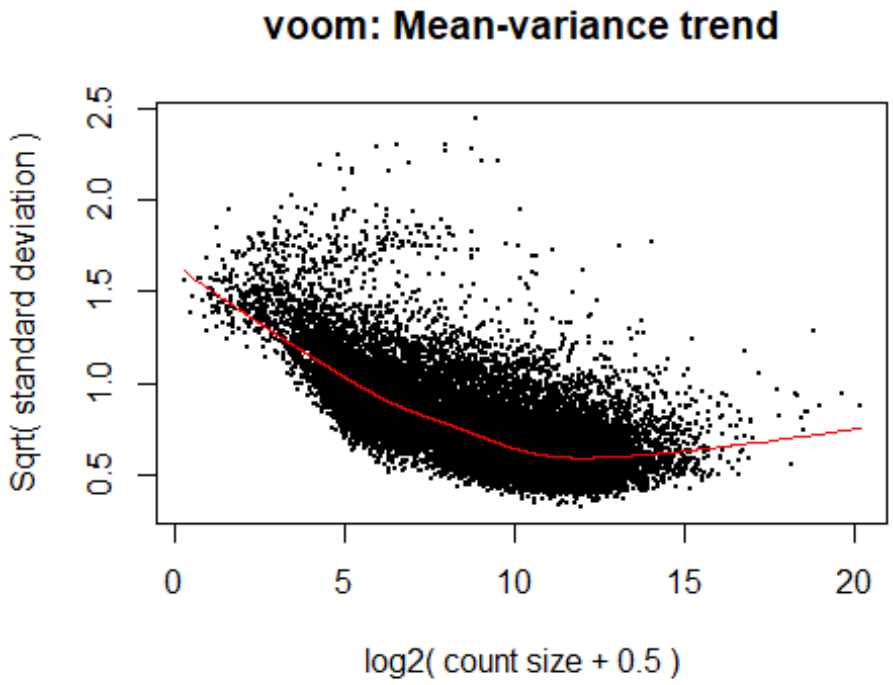



```
#Matriz de diseño
madis <- cbind(ELI =
c(1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0),
               NIT =
c(0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0),
               SFI=
c(0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1))
rownames(madis) <- (rownames(millionlog@colData))
madis
```

```
##       ELI NIT SFI
## ELI1    1   0   0
## ELI2    1   0   0
## ELI3    1   0   0
## ELI4    1   0   0
## ELI5    1   0   0
## ELI6    1   0   0
## ELI7    1   0   0
## ELI8    1   0   0
## ELI9    1   0   0
## ELI10   1   0   0
## NIT1    0   1   0
## NIT2    0   1   0
## NIT3    0   1   0
## NIT4    0   1   0
## NIT15   0   1   0
## NIT6    0   1   0
```

```
## NIT7    0   1   0
## NIT8    0   1   0
## NIT9    0   1   0
## NIT10   0   1   0
## SFI1    0   0   1
## SFI2    0   0   1
## SFI3    0   0   1
## SFI4    0   0   1
## SFI5    0   0   1
## SFI6    0   0   1
## SFI7    0   0   1
## SFI8    0   0   1
## SFI9    0   0   1
## SFI10   0   0   1
```

```
trans <- as.DGEList(millionlist)
transnorm <- calcNormFactors(millionlist)
transvoom <- voom(transnorm,madis,plot = TRUE)
```



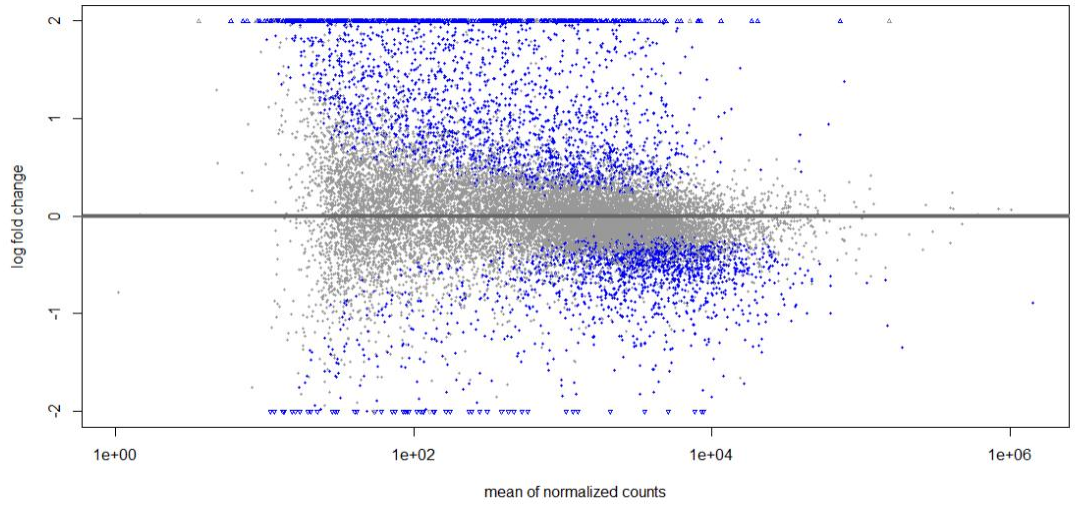

```
#Análisis de expresión diferencial con función DESeq
dea <- DESeq(millionlist)

## estimating size factors

## estimating dispersions

## gene-wise dispersion estimates
```

```
## mean-dispersion relationship

## final dispersion estimates

## fitting model and testing

## -- replacing outliers and refitting for 111 genes
## -- DESeq argument 'minReplicatesForReplace' = 7
## -- original counts are preserved in counts(dds)

## estimating dispersions

## fitting model and testing

resc1 <- results(dea,contrast = c("group", "ELI", "SFI"))
resc2 <- results(dea,contrast = c("group", "ELI", "NIT"))
resc3 <- results(dea,contrast = c("group", "SFI", "NIT"))
head(resc1)

## log2 fold change (MLE): group ELI vs SFI
## Wald test p-value: group ELI vs SFI
## DataFrame with 6 rows and 6 columns
##                     baseMean log2FoldChange     lfcSE      stat
pvalue
##                    <numeric>      <numeric> <numeric> <numeric>
<numeric>
## ENSG00000227232.4  801.9297     -0.0827186  0.207520 -0.398604
0.690185
## ENSG00000233750.3   20.3061      0.2158947  0.506793  0.426002
0.670106
## ENSG00000237683.5  828.8481      0.0877455  0.436604  0.200973
0.840720
## ENSG00000241860.2   86.5393      0.4904024  0.341629  1.435482
0.151150
## ENSG00000228463.4   50.0977      0.2845788  0.476436  0.597308
0.550302
## ENSG00000237094.7   43.6307      0.0513586  0.373789  0.137400
0.890715
##                        padj
##                   <numeric>
## ENSG00000227232.4  0.828875
## ENSG00000233750.3  0.815539
## ENSG00000237683.5  0.916326
## ENSG00000241860.2  0.342823
## ENSG00000228463.4  0.732234
## ENSG00000237094.7  0.945190

#Ordeno según pvalor
resOrdered1 <- resc1[order(resc1$pvalue),]
resOrdered2 <- resc2[order(resc2$pvalue),]
resOrdered3 <- resc3[order(resc3$pvalue),]
```

```r
#Número de genes diferenciados con alpha 0.01 en resc1
summary(resc1)

##
## out of 19537 with nonzero total read count
## adjusted p-value < 0.1
## LFC > 0 (up)       : 2530, 13%
## LFC < 0 (down)     : 1725, 8.8%
## outliers [1]       : 0, 0%
## low counts [2]     : 0, 0%
## (mean count < 1)
## [1] see 'cooksCutoff' argument of ?results
## [2] see 'independentFiltering' argument of ?results

#Número de genes diferenciados con alpha 0.01 en resc2
summary(resc2)

##
## out of 19537 with nonzero total read count
## adjusted p-value < 0.1
## LFC > 0 (up)       : 2603, 13%
## LFC < 0 (down)     : 1207, 6.2%
## outliers [1]       : 0, 0%
## low counts [2]     : 0, 0%
## (mean count < 1)
## [1] see 'cooksCutoff' argument of ?results
## [2] see 'independentFiltering' argument of ?results

#Número de genes diferenciados con alpha 0.01 en resc3
summary(resc3)

##
## out of 19537 with nonzero total read count
## adjusted p-value < 0.1
## LFC > 0 (up)       : 193, 0.99%
## LFC < 0 (down)     : 70, 0.36%
## outliers [1]       : 0, 0%
## low counts [2]     : 0, 0%
## (mean count < 1)
## [1] see 'cooksCutoff' argument of ?results
## [2] see 'independentFiltering' argument of ?results

#Número de genes diferenciados con alpha 0.05 en resc1
summary(resc1, alpha=0.05)

##
## out of 19537 with nonzero total read count
## adjusted p-value < 0.05
## LFC > 0 (up)       : 1943, 9.9%
## LFC < 0 (down)     : 1071, 5.5%
## outliers [1]       : 0, 0%
## low counts [2]     : 0, 0%
```

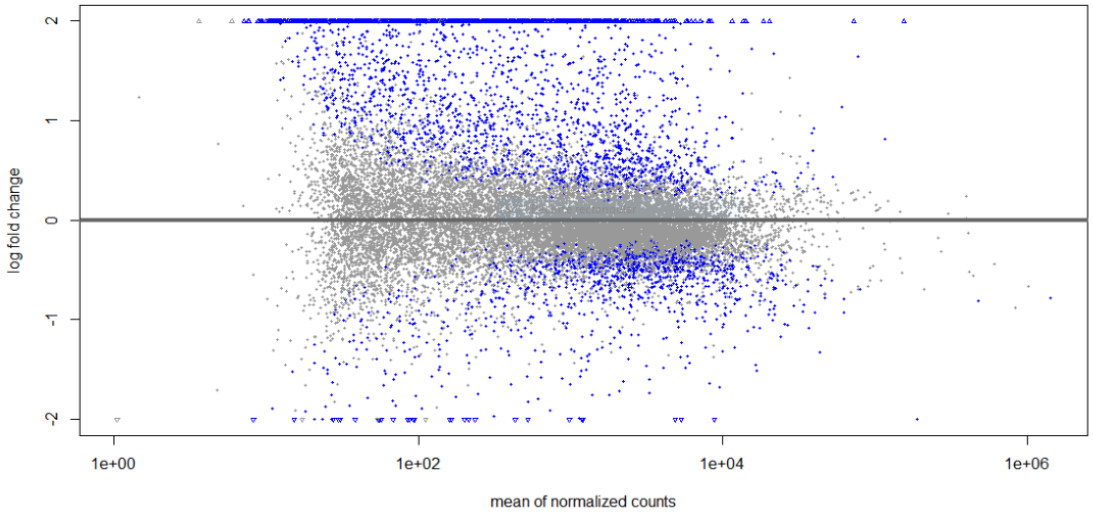
```
## (mean count < 1)
## [1] see 'cooksCutoff' argument of ?results
## [2] see 'independentFiltering' argument of ?results

#Número de genes diferenciados con alpha 0.05 en resc2
summary(resc2, alpha=0.05)

##
## out of 19537 with nonzero total read count
## adjusted p-value < 0.05
## LFC > 0 (up)       : 2139, 11%
## LFC < 0 (down)     : 653, 3.3%
## outliers [1]       : 0, 0%
## low counts [2]     : 0, 0%
## (mean count < 1)
## [1] see 'cooksCutoff' argument of ?results
## [2] see 'independentFiltering' argument of ?results

#Número de genes diferenciados con alpha 0.05 en resc3
summary(resc3, alpha=0.05)

##
## out of 19537 with nonzero total read count
## adjusted p-value < 0.05
## LFC > 0 (up)       : 81, 0.41%
## LFC < 0 (down)     : 17, 0.087%
## outliers [1]       : 0, 0%
## low counts [2]     : 0, 0%
## (mean count < 1)
## [1] see 'cooksCutoff' argument of ?results
## [2] see 'independentFiltering' argument of ?results

#Gráfico resc1 expresión diferencial
plotMA(resc1, ylim=c(-2,2))
```

```
knitr::include_graphics("C:/Bea/Master/Datos omicos
Bea/PEC2/Analisis/Results/figure7.png")
```



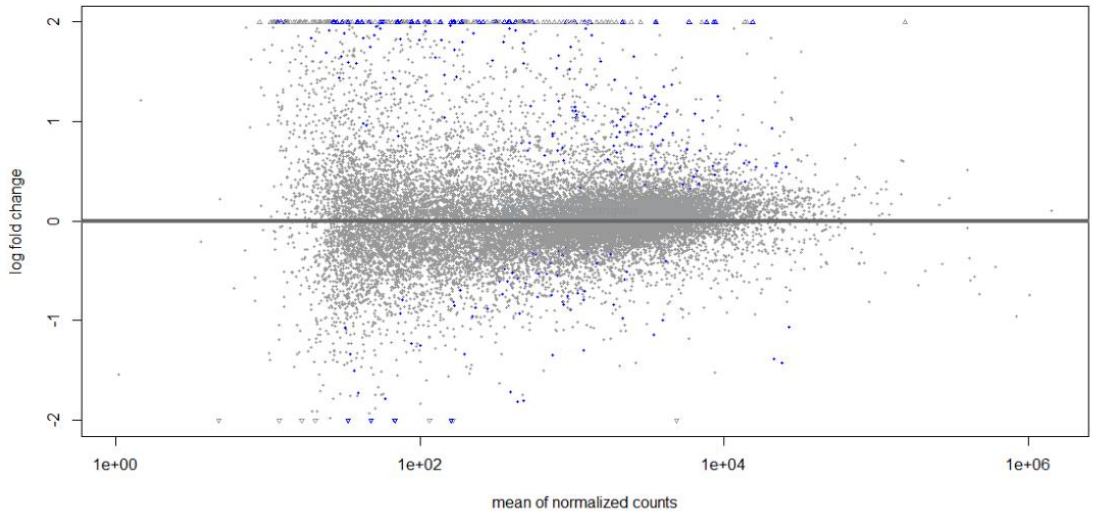

```
#Gráfico resc2 expresión diferencial
plotMA(resc2, ylim=c(-2,2))
```

```
knitr::include_graphics("C:/Bea/Master/Datos omicos
Bea/PEC2/Analisis/Results/figure8.png")
```



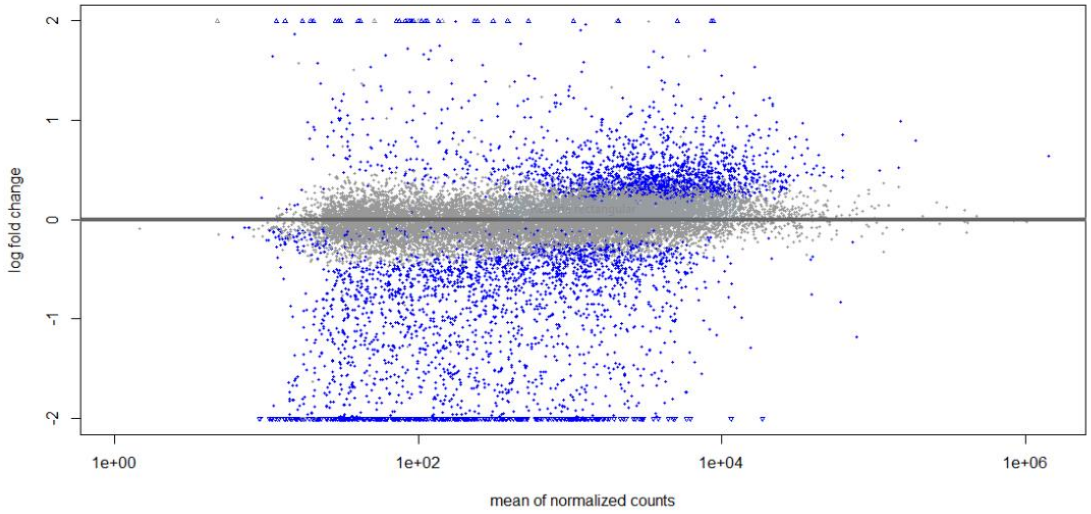

```
#Gráfico resc3 expresión diferencial
plotMA(resc3, ylim=c(-2,2))
```

```
knitr::include_graphics("C:/Bea/Master/Datos omicos
Bea/PEC2/Analisis/Results/figure9.png")
```



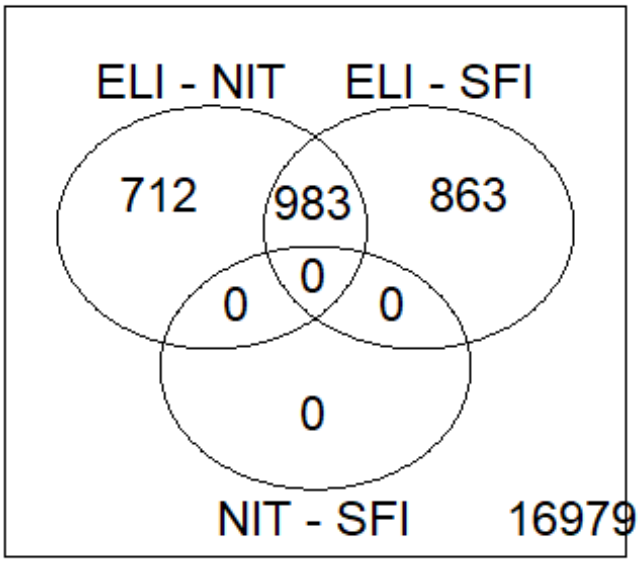

```
#Reducción con función lfcShrink
resc1LFC <- lfcShrink(dea, coef="group_SFI_vs_ELI", type="apeglm")

## using 'apeglm' for LFC shrinkage. If used in published research,
please cite:
##     Zhu, A., Ibrahim, J.G., Love, M.I. (2018) Heavy-tailed prior
distributions for
##     sequence count data: removing the noise and preserving large
differences.
##     Bioinformatics. https://doi.org/10.1093/bioinformatics/bty895

#Gráfico resc1 con reducción LFC
plotMA(resc1LFC, ylim=c(-2,2))
```

```
knitr::include_graphics("C:/Bea/Master/Datos omicos
Bea/PEC2/Analisis/Results/figure10.png")
```

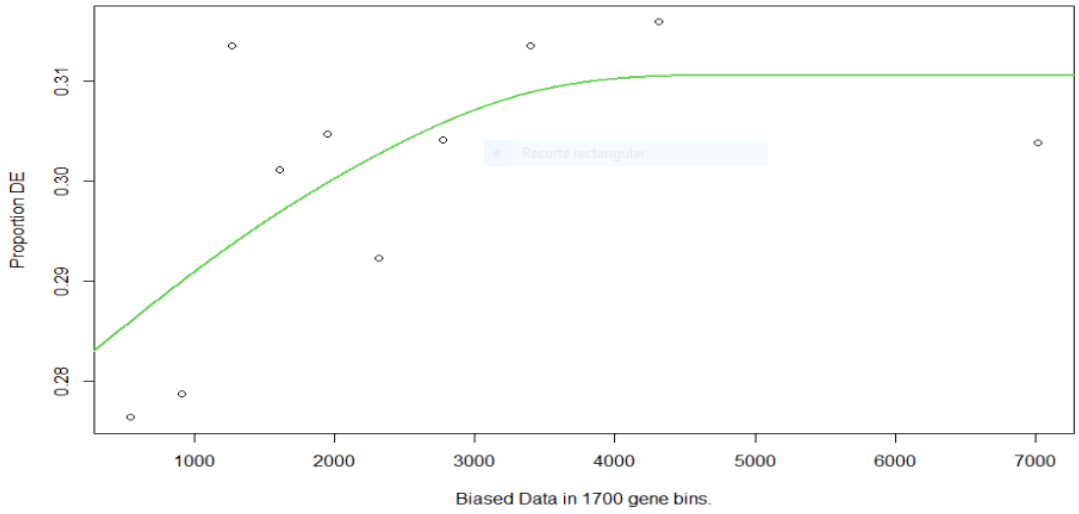



```
#Anotación de genes resc1 (symbol)
tmp=gsub("\\..*","",row.names(resc1))
resc1$symbol <- mapIds(org.Hs.eg.db,
                       keys=tmp,
                       column="SYMBOL",
                       keytype="ENSEMBL",
                       multiVals="first")

## 'select()' returned 1:many mapping between keys and columns

#Anotación de genes resc1 (entrez)
resc1$entrez <- mapIds(org.Hs.eg.db,
                       keys=tmp,
                       column="ENTREZID",
                       keytype="ENSEMBL",
                       multiVals="first")

## 'select()' returned 1:many mapping between keys and columns

#Anotación de genes resc2 (symbol)
tmp=gsub("\\..*","",row.names(resc2))
resc2$symbol <- mapIds(org.Hs.eg.db,
                       keys=tmp,
                       column="SYMBOL",
                       keytype="ENSEMBL",
                       multiVals="first")

## 'select()' returned 1:many mapping between keys and columns

#Anotación de genes resc2 (entrez)
resc2$entrez <- mapIds(org.Hs.eg.db,
                       keys=tmp,
                       column="ENTREZID",
                       keytype="ENSEMBL",
                       multiVals="first")
```

```
## 'select()' returned 1:many mapping between keys and columns

#Anotación de genes resc3 (symbol)
tmp=gsub("\\..*","",row.names(resc3))
resc3$symbol <- mapIds(org.Hs.eg.db,
                    keys=tmp,
                    column="SYMBOL",
                    keytype="ENSEMBL",
                    multiVals="first")

## 'select()' returned 1:many mapping between keys and columns

#Anotación de genes resc3 (entrez)
resc3$entrez <- mapIds(org.Hs.eg.db,
                    keys=tmp,
                    column="ENTREZID",
                    keytype="ENSEMBL",
                    multiVals="first")

## 'select()' returned 1:many mapping between keys and columns

#Matriz de contraste
macont <- makeContrasts(ELI - NIT, ELI - SFI, NIT -SFI, levels= madis)
macont

##       Contrasts
## Levels ELI - NIT ELI - SFI NIT - SFI
##    ELI         1         1         0
##    NIT        -1         0         1
##    SFI         0        -1        -1

#Realizo contraste con lmFit
fit <- lmFit(transvoom)
fit.cont <- contrasts.fit(fit, macont)
fit.cont <- eBayes(fit.cont)

#Genes diferencialmente expresados
summa.fit <- decideTests(fit.cont,adjust.method = "fdr")
summary(summa.fit)

##        ELI - NIT ELI - SFI NIT - SFI
## Down         343       507         0
## NotSig     17842     17691     19537
## Up          1352      1339         0

vennDiagram(summa.fit)
```



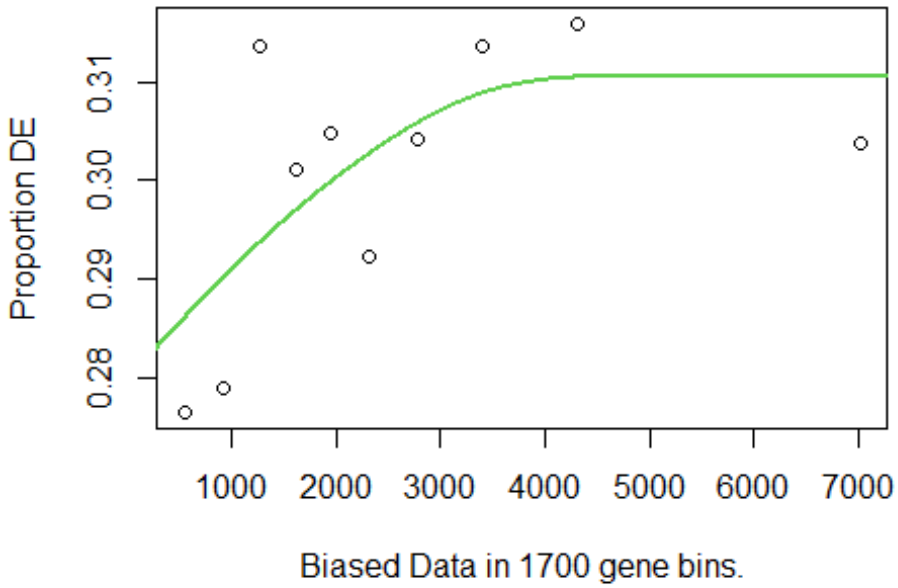

```
knitr::include_graphics("C:/Bea/Master/Datos omicos
Bea/PEC2/Analisis/Results/figure12.png")
```




```
knitr::include_graphics("C:/Bea/Master/Datos omicos
Bea/PEC2/Analisis/Results/figure13.png")
```

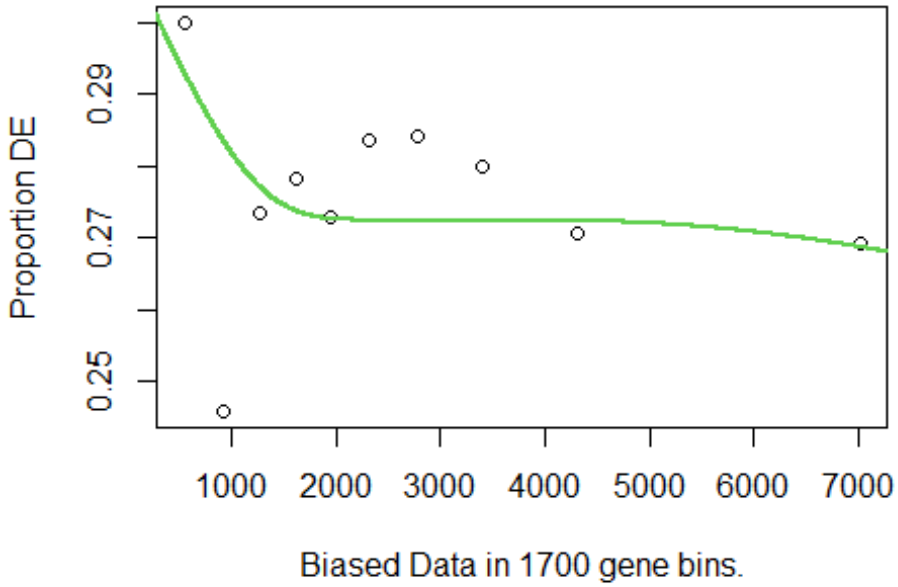
	category	over_represented_pvalue	under_represented_pvalue	numDEInCat	numInCat
	<chr>	<dbl>	<dbl>	<int>	<int>
3475	GO:0006955	5.484520e-39	1	740	1660
1004	GO:0002376	2.370685e-38	1	1028	2478
13419	GO:0050896	7.833587e-36	1	2523	7143
7022	GO:0023052	4.564483e-34	1	1885	5119
3572	GO:0007154	1.413214e-33	1	1888	5137
3583	GO:0007165	2.244684e-32	1	1737	4686

```
#Análisis de significación biológica resc1
genes1 <-
as.integer(p.adjust(resc1@listData$pvalue[resc1@listData$log2FoldChange!=
0],method="BH")<.05)
names(genes1) <- row.names(resc1@rownames)
genesna1 <- na.omit(genes1)
DEgenes1 <- as.integer(resc1$pvalue <= 0.05)
tmp1 <- gsub("\\..*","",row.names(resc1))
names(DEgenes1) <- tmp1
pwf1 <-nullp(DEgenes1,"hg19","ensGene")

## Loading hg19 length data...
```




```
GO.wall1 <- goseq(pwf1,"hg19","ensGene")

## Fetching GO annotations...

## For 4507 genes, we could not find any categories. These genes will be
excluded.

## To force their use, please run with use_genes_without_cat=TRUE (see
documentation).
```

```
## This was the default behavior for version 1.15.1 and earlier.

## Calculating the p-values...

## 'select()' returned 1:1 mapping between keys and columns

head(GO.wall1)

##         category over_represented_pvalue under_represented_pvalue
numDEInCat
## 3475  GO:0006955           5.484520e-39                         1
740
## 1004  GO:0002376           2.370685e-38                         1
1028
## 13419 GO:0050896           7.833587e-36                         1
2523
## 7022  GO:0023052           4.564483e-34                         1
1885
## 3572  GO:0007154           1.413214e-33                         1
1888
## 3583  GO:0007165           2.244684e-32                         1
1737
##       numInCat                  term ontology
## 3475      1660       immune response       BP
## 1004      2478 immune system process       BP
## 13419     7143  response to stimulus       BP
## 7022      5119             signaling       BP
## 3572      5137    cell communication       BP
## 3583      4686   signal transduction       BP

enrichedGO1 <-
GO.wall1$category[p.adjust(GO.wall1$over_represented_pvalue,
method="BH")<0.05]
head(enrichedGO1)

## [1] "GO:0006955" "GO:0002376" "GO:0050896" "GO:0023052" "GO:0007154"
## [6] "GO:0007165"

#Análisis de significación biológica resc2
genes2 <-
as.integer(p.adjust(resc2@listData$pvalue[resc2@listData$log2FoldChange!=
0],method="BH")<.05)
names(genes2) <- row.names(resc2@rownames)
genesna2 <- na.omit(genes2)
DEgenes2 <- as.integer(resc2$pvalue <= 0.05)
tmp2 <- gsub("\\..*","",row.names(resc2))
names(DEgenes2) <- tmp2
pwf2 <-nullp(DEgenes2,"hg19","ensGene")

## Loading hg19 length data...
```



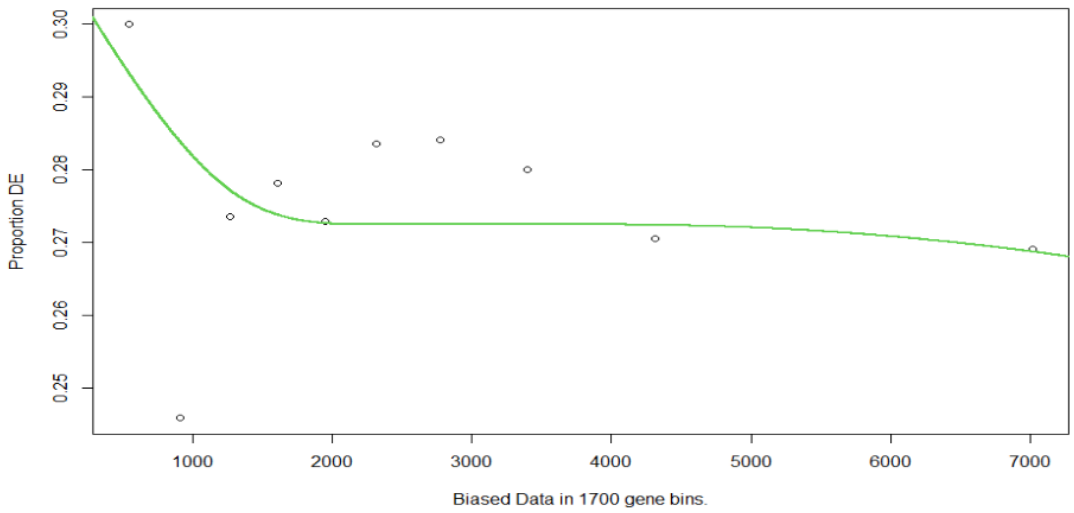

```
GO.wall2 <- goseq(pwf2,"hg19","ensGene")

## Fetching GO annotations...

## For 4507 genes, we could not find any categories. These genes will be
excluded.

## To force their use, please run with use_genes_without_cat=TRUE (see
documentation).

## This was the default behavior for version 1.15.1 and earlier.

## Calculating the p-values...

## 'select()' returned 1:1 mapping between keys and columns

head(GO.wall2)

##         category over_represented_pvalue under_represented_pvalue
numDEInCat
## 1004  GO:0002376           3.969272e-53                         1
1000
## 3475  GO:0006955           4.057530e-53                         1
729
## 918   GO:0002250           2.925859e-41                         1
221
## 12437 GO:0046649           2.117268e-40                         1
313
```

```
## 1174  GO:0002682            4.326224e-40                          1
555
## 11821 GO:0045321            1.063764e-38                          1
494
##       numInCat                                    term ontology
## 1004      2478                immune system process       BP
## 3475      1660                      immune response       BP
## 918        364             adaptive immune response       BP
## 12437      594                lymphocyte activation       BP
## 1174      1262 regulation of immune system process       BP
## 11821     1097                 leukocyte activation       BP
```

```
enrichedGO2 <-
GO.wall2$category[p.adjust(GO.wall2$over_represented_pvalue,
method="BH")<0.05]
head(enrichedGO2)
```

```
## [1] "GO:0002376" "GO:0006955" "GO:0002250" "GO:0046649" "GO:0002682"
## [6] "GO:0045321"
```

```
knitr::include_graphics("C:/Bea/Master/Datos omicos
Bea/PEC2/Analisis/Results/figure14.png")
```

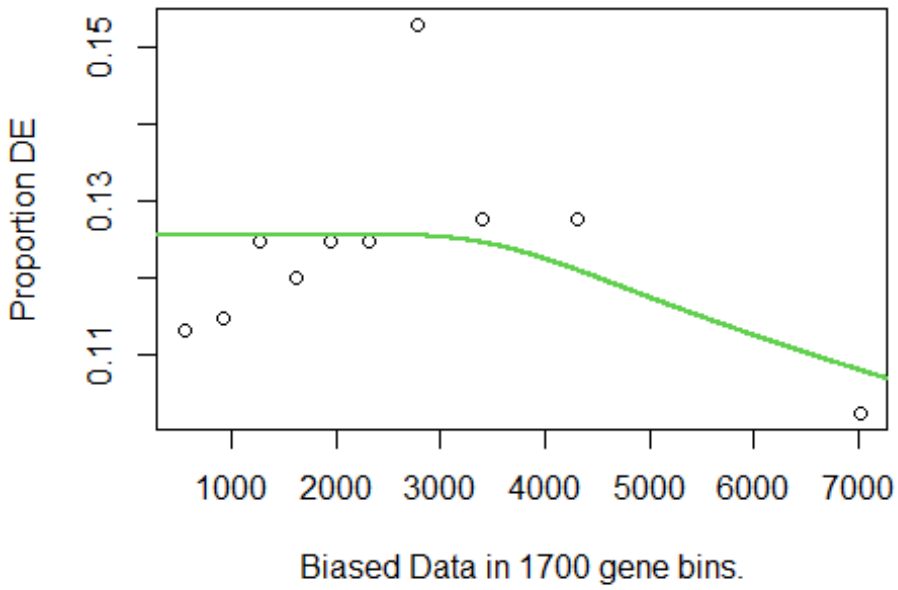



```
knitr::include_graphics("C:/Bea/Master/Datos omicos
Bea/PEC2/Analisis/Results/figure15.png")
```

	category <chr>	over_represented_pvalue <dbl>	under_represented_pvalue <dbl>	numDEInCat <int>	numInCat <int>
1004	GO:0002376	3.969272e-53	1	1000	2478
3475	GO:0006955	4.057530e-53	1	729	1660
918	GO:0002250	2.925859e-41	1	221	364
12437	GO:0046649	2.117268e-40	1	313	594
1174	GO:0002682	4.326224e-40	1	555	1262
11821	GO:0045321	1.063764e-38	1	494	1097

```
#Análisis de significación biológica resc3
genes3 <-
as.integer(p.adjust(resc3@listData$pvalue[resc3@listData$log2FoldChange!=
```

```
0],method="BH")<.05)
names(genes3) <- row.names(resc3@rownames)
genesna3 <- na.omit(genes3)
DEgenes3 <- as.integer(resc3$pvalue <= 0.05)
tmp3 <- gsub("\\..*","",row.names(resc3))
names(DEgenes3) <- tmp3
pwf3 <-nullp(DEgenes3,"hg19","ensGene")

## Loading hg19 length data...
```



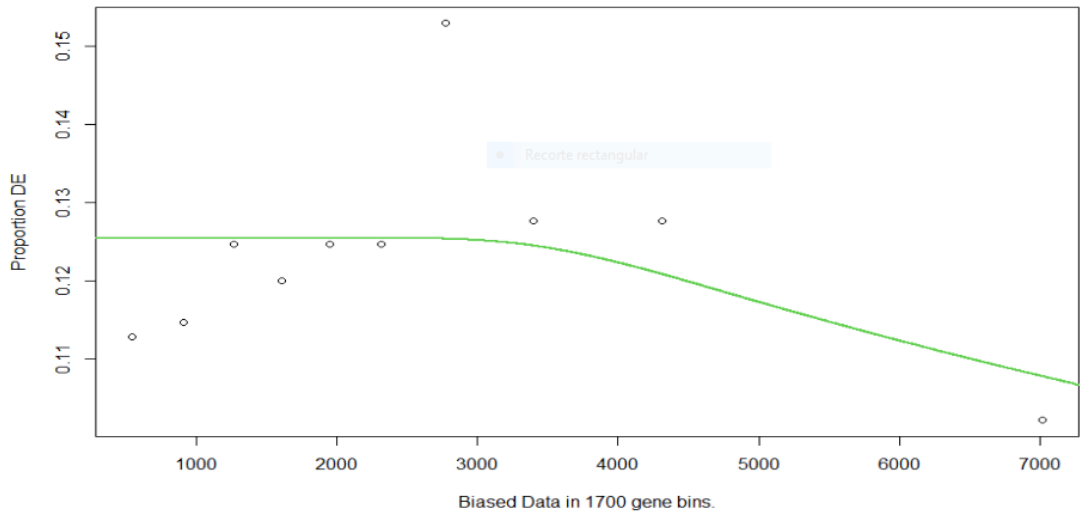

```
GO.wall3 <- goseq(pwf3,"hg19","ensGene")

## Fetching GO annotations...

## For 4507 genes, we could not find any categories. These genes will be
excluded.

## To force their use, please run with use_genes_without_cat=TRUE (see
documentation).

## This was the default behavior for version 1.15.1 and earlier.

## Calculating the p-values...

## 'select()' returned 1:1 mapping between keys and columns

head(GO.wall3)
```

```
##         category over_represented_pvalue under_represented_pvalue
numDEInCat
## 6968  GO:0022610            2.508950e-13                         1
238
## 3573  GO:0007155            4.668833e-13                         1
236
## 7115  GO:0030155            1.798133e-10                         1
130
## 918   GO:0002250            7.453122e-10                         1
87
## 17755 GO:0098609            2.677545e-09                         1
144
## 3577  GO:0007159            7.131923e-09                         1
73
##       numInCat                          term ontology
## 6968      1245          biological adhesion       BP
## 3573      1239                 cell adhesion       BP
## 7115       616  regulation of cell adhesion       BP
## 918        364     adaptive immune response       BP
## 17755      732           cell-cell adhesion       BP
## 3577       300 leukocyte cell-cell adhesion       BP
```

```
enrichedGO3 <-
GO.wall3$category[p.adjust(GO.wall3$over_represented_pvalue,
method="BH")<0.05]
head(enrichedGO3)
```

```
## [1] "GO:0022610" "GO:0007155" "GO:0030155" "GO:0002250" "GO:0098609"
## [6] "GO:0007159"
```

```
knitr::include_graphics("C:/Bea/Master/Datos omicos
Bea/PEC2/Analisis/Results/figure16.png")
```



```
knitr::include_graphics("C:/Bea/Master/Datos omicos Bea/PEC2/Analisis/Results/figure17.png")
```

	category <chr>	over_represented_pvalue <dbl>	under_represented_pvalue <dbl>	numDEInCat <int>	numInCat <int>
6968	GO:0022610	2.508950e-13	1	238	1245
3573	GO:0007155	4.668833e-13	1	236	1239
7115	GO:0030155	1.798133e-10	1	130	616
918	GO:0002250	7.453122e-10	1	87	364
17755	GO:0098609	2.677545e-09	1	144	732
3577	GO:0007159	7.131923e-09	1	73	300