PEC 2 - Análisis de datos Ómicos

Alberto Castillo Aroca

11/6/2020

Table of Contents

1. Abstract

En el siguiente estudio se realiza un análisis de expresión diferenciada (RNA-seq) con datos de tejidos de tiroides de tres tipos: Not infiltrated tissues (NIT), Small focal infiltrates (SFI), Extensive Lymphoid Infiltrates (ELI).

Para la investigación se seleccionaron 10 muestras aleatorias de cada tipo desde una base de datos con 292 muestras totales. Posteriormente se realizó un análisis por pares, comparando NIT vs SFI, NIT vs ELI y SFI vs ELI.

Repositorio de GitHub:

https://github.com/alcastaro/PEC2_OMICAS_ACA

2. Objetivos

El objetivo del estudio fue identificar patrones diferenciados de expresión génica entre las muestras, específicamente genes sobre y sub expresados.

3. Materiales y métodos

3.1 Naturaleza de los datos

Para la investigación se utilizó una base de datos procedente del Genotype-Tissue Expression (GTEx), los cuales fueron analizados con el software R 4.0 y la versión 3.11 de BiocManager / Bioconductor.

Con estas herramientas se aplicaron métodos de pre-filtrado de los datos, estabilización de la varianza, análisis de conglomerados génicos, PCA, MDS y el análisis de expresión diferencial con el paquete DESeq2. Finalmente se procedió a visualizar los resultados y eliminar los efectos ocultos por bloques.

3.2 Procedimiento general

Para llevar a cabo el análisis se hicieron dos grandes procesos, el primero fue un análisis exploratorio con los datos estandarizados y la varianza estabilizada, posteriormente se utilizaron los datos brutos para realizar el análisis de expresión diferencial.

A continuación se describe el proceso general del análisis con una comparación entre NIT vs SFI, para posteriormente analizar los resultados de las tres comparaciones en el siguiente capítulo.

Naturaleza, importación y formato de los datos

En este apartado es importante destacar que los datos utilizados constan de una matriz con información general de las muestras analizadas tales como el tipo de experimento, nombre de la muestra, grupo de análisis, tipo molecular, sexo de la persona analizada, entre otros.

Por otra parte, se cuenta con una matriz de conteo que consiste en un conjunto de filas que refieren a códigos de ENSEMBL y un conjunto de columnas referentes a las muestras analizadas.

Sin embargo, que los códigos de ENSEMBL cuentan con un sufijo referente a la versión del mismo, lo cual dificulta su anotación posterior. Por esta razón se procedió a eliminar dicho sufijo.

```
library(readr)
targets <- read_delim("targets.csv", ";",
    escape_double = FALSE, trim_ws = TRUE)

counts <- read_delim("counts.csv", ";",
    escape_double = FALSE, trim_ws = TRUE)

counts=as.data.frame(counts)
Ensembl = gsub("\\..*", "", counts$X1, fixed = FALSE)
rownames(counts)=Ensembl
```

Con el fin de contar con un panel balanceado, se procedió a seleccionar 10 muestras de cada tipo según la matriz de características, con lo cual posteriormente se segmentó la matriz de conteo.

```
#Observando los grupos
table(targets$Group,targets$Grupo_analisis)
```

```
       1   2   3
  ELI   0   0  14
  NIT 236   0   0
  SFI   0  42   0
```

```
set.seed(12345)
m.eli=sample(1:14, 10, replace=F)
m.nit=sample(1:236, 10, replace=F)
m.sfi=sample(1:42, 10, replace=F)

m.eli=targets[targets$Group=="ELI",][m.eli,]
m.nit=targets[targets$Group=="NIT",][m.nit,]
m.sfi=targets[targets$Group=="SFI",][m.sfi,]
```

En esta explicación del proceso se utilizaron los valores de las muestras de SFI/NIT y se utilizó la función DESeqDataSetFromMatrix para crear un objeto DESeqDataSet que permitiera ser analizado con el paquete DESeq2.

```
data.targets=rbind(m.sfi,m.nit)
data.counts=counts[,data.targets$Sample_Name]

data.targets$Group=factor(data.targets$Group)

dds <- DESeqDataSetFromMatrix(countData = data.counts,
                              colData = data.targets,
                              design = ~ Group)

converting counts to integer mode
```

Filtrado

Cabe destacar no todas las filas poseen conteos, por esta razón y para optimizar memoria y costo de cómputo, se decidió hacer un prefiltrado, eliminando todas las filas que tuvieran 1 o menos conteos.

```
dds <- dds[ rowSums(counts(dds)) > 1, ]
```

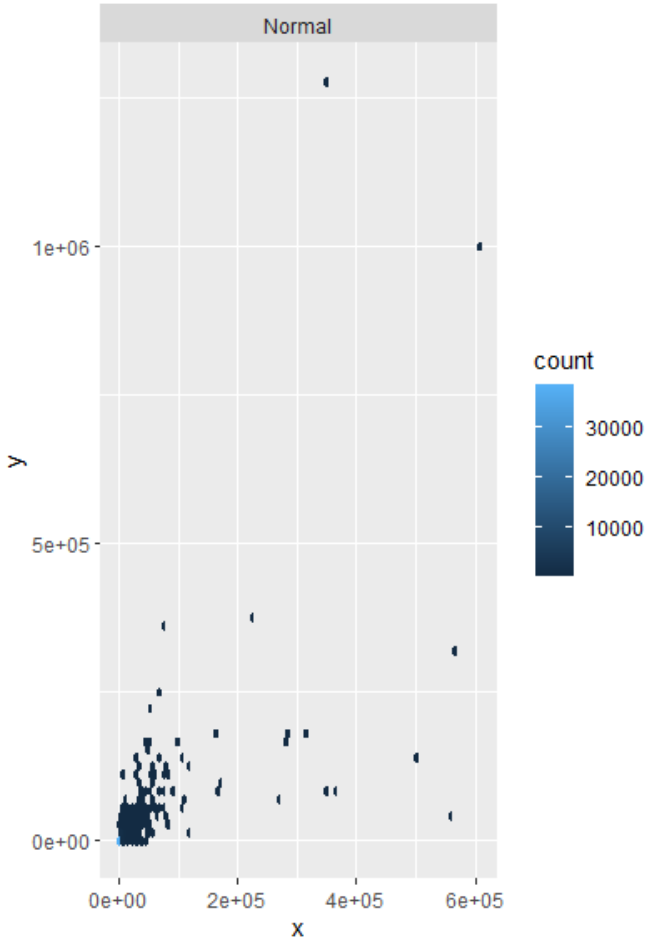
Análisis Exploratorio

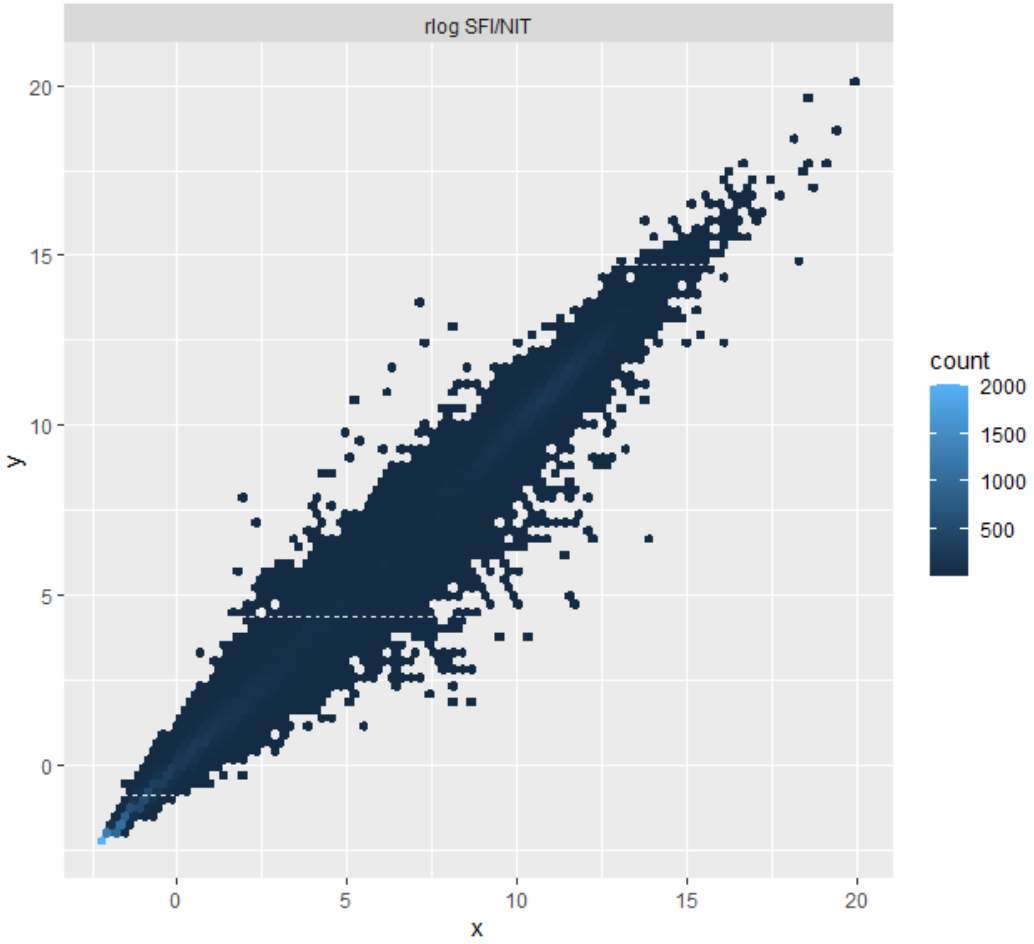
Estabilización de la varianza

Con el fin de realizar un análisis exploratorio de los datos se decidió realizar un ajuste de la varianza, ya que el PCA y MDS funcionan mejor en escenarios e homocedasticidad. En vista de que la base de datos es pequeña (n<30), se hizo uso del método rlog debido a que este funciona mejor con un número reducido de muestras.

```
rld <- rlog(dds, blind = FALSE)
```

A continuación se presenta un diagrama de dispersión entre los datos de la primera y la segunda muestra de la base sin estabilización de la varianza y tras el proceso rlog, con lo cual se evidencia que los datos transformados presentan homocedasticidad.

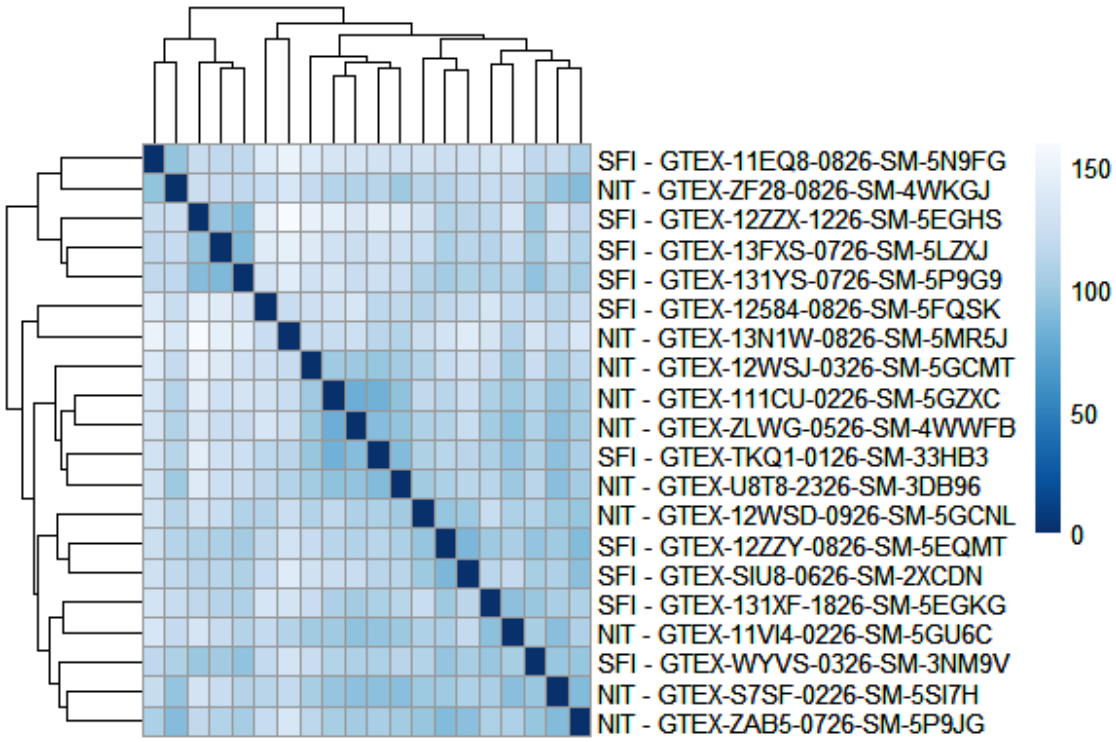

Normal
1e+06
5e+05
0e+00
y
0e+00
2e+05
4e+05
6e+05
x
count
30000
20000
10000



Similitud entre las muestras

Con estos datos homocedásticos se procedió a calcular las distancias entre las muestras para realizar un pheatmap que permite visualizar las muestras más similares y diferentes.

Grosso modo se evidencia que existe mayor similitud entre las muestras de un mismo tipo, ya sea NIT o SFI.

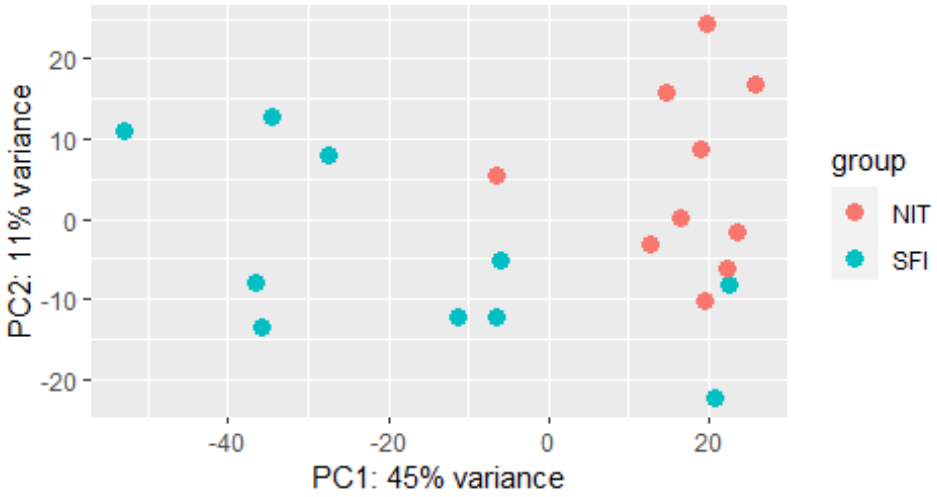


PCA Plot

Con el PCA plot de los datos con varianza estabilizada se puede observar la diferenciación de los tipos de muestras ya que este análisis maximiza la varianza entre estas y permite observar las diferencias existentes.

El resultado demuestra que sí existen diferencias entre ambos tipos de muestras, por lo cual se puede presumir a priori que existe un patrón de expresión diferenciado.

```
plotPCA(rld, intgroup = c("Group"))
```

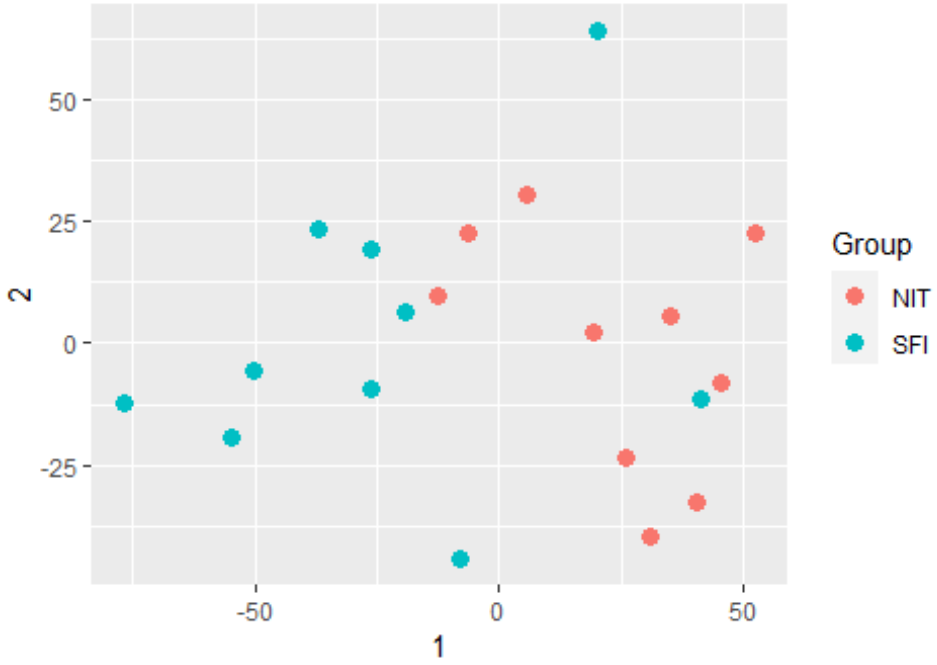


MDS Plot

También se hizo un gráfico de Multi-Dimensional Scaling (MDS) que permite tener una visualización de los patrones de similaridad entre los genes, para esto se dispuso de una matriz de distancias con la cual se obtiene un gráfico donde las distancias entre los puntos (genes) es equivalente a la similaridad.

En este sentido, se observa que existe un agrupamiento de los genes de cada tipo de tejido, esto significa que existen similitudes intragrupales y diferencias extra-grupales, lo cual indica que se puede realizar un análisis de expresión diferencial.

```
mds <- as.data.frame(colData(rld))  %>%
         cbind(cmdscale(sampleDistMatrix))
ggplot(mds, aes(x = `1`, y = `2`, color = Group)) +
  geom_point(size = 3) + coord_fixed()
```



Expresión diferencial

En vista de que se confirmó que existen diferencias entre los tipos de muestras, se procedió a realizar un análisis de expresión diferencial con los datos brutos y el paquete DESeq2.

Cabe la pena recordar que en este paso se deben utilizar los datos brutos para que las pruebas de hipótesis funcionen correctamente.

En este paso se creó un nuevo objeto de tipo DESeqDataSet denominado ddsdeseq con base en los datos brutos dds. Posteriormente se aplicó la función results especificando que el contraste se hace entre los valores de la variable Group, específicamente entre NIT y SFI. En este sentido, NIT sería el numerador y SFI el denominador en la comparación.

Este proceso incluye la estimación o uso de los tamaños pre-existentes de los factores, estimar la dispersión y utilizar la Negative Binomial GLM y utilizar el test de Wald para calcular los p-valores.

El resultado es un objeto DESeqResults que posee el listado de códigos ENSEMBL, el bseMean, el Log2FoldChange, error estándar (lfcSE), el Test de Wald (stat), p-valor y p-valor ajustado por el método BH.

```
ddsdeseq <- DESeq(dds, parallel =TRUE)
res <- results(ddsdeseq, contrast=c("Group","SFI","NIT"))
res
```

```
log2 fold change (MLE): Group SFI vs NIT
Wald test p-value: Group SFI vs NIT
DataFrame with 41302 rows and 6 columns
                   baseMean log2FoldChange     lfcSE       stat    pvalue
                  <numeric>      <numeric> <numeric>  <numeric> <numeric>
ENSG00000223972    2.930479      0.1635475  0.582452  0.2807911  0.778871
ENSG00000227232  872.360023      0.1516108  0.231579  0.6546820  0.512672
ENSG00000243485    1.556065     -0.6841320  0.787503 -0.8687358  0.384992
ENSG00000237613    1.291786      0.0289701  0.933060  0.0310485  0.975231
ENSG00000268020    0.335797      0.7560211  1.616585  0.4676654  0.640024
...                     ...            ...       ...        ...       ...
ENSG00000198695 6.90349e+04     -0.7796180  0.734589 -1.0612981  0.288554
ENSG00000210194 3.87846e+01      0.0386472  1.032479  0.0374315  0.970141
ENSG00000198727 4.20140e+05     -0.3483085  0.319120 -1.0914644  0.275069
ENSG00000210195 7.26932e-01      0.3049961  1.110975  0.2745301  0.783677
ENSG00000210196 1.55836e+00     -1.2542654  0.958611 -1.3084197  0.190731
                     padj
                <numeric>
ENSG00000223972  0.951152
ENSG00000227232  0.852350
ENSG00000243485        NA
ENSG00000237613        NA
ENSG00000268020        NA
...                   ...
ENSG00000198695  0.719636
ENSG00000210194  0.992854
ENSG00000198727  0.710909
ENSG00000210195        NA
ENSG00000210196        NA
```

Este objeto permite identificar que de las 41299 observaciones, hay 166 (0.4%) sobre expresadas (up) y 834 (2%) subexpresadas, a un nivel de confianza de p-value<0.1

```
summary(res)
```

```


out of 41299 with nonzero total read count
adjusted p-value < 0.1
LFC > 0 (up)       : 834, 2%
LFC < 0 (down)     : 166, 0.4%
outliers [1]       : 0, 0%
low counts [2]     : 12814, 31%
(mean count < 2)
```

```
[1] see 'cooksCutoff' argument of ?results
[2] see 'independentFiltering' argument of ?results
```

Por otra parte, si se requiere ajustar más el criterio y modificar el False Discovery Rate al 5%, sólo 711 observaciones estarían diferencialmente expresadas.

```
res.05 <- results(ddsdeseq, alpha = 0.05)
table(res.05$padj < 0.05)
```

```
FALSE  TRUE
29378   711
```

En cambio, si se toma sólamente el 5% de los valores del False Discovery Rate como valores significativos, la cantidad de genes diferencialmente expresados se ubica en 719.

```
sum(res$padj < 0.05, na.rm=TRUE)
```

```
[1] 719
```

Sin embargo, se mantendrá un valor de 0.1, debido a que este es el default del análisis, para crear una tabla con los valores diferencialmente expresados ordenados de acuerdo con el Log2FoldChange. Concretamente se presentarán los 5 más sub-regulados.

Nota: cabe recordar que el Log2FoldChange permite evaluar si un gen está sobre o sub regulado.

```
resSig <- subset(res, padj < 0.1)
head(resSig[ order(resSig$log2FoldChange), ])
```

```
log2 fold change (MLE): Group SFI vs NIT
Wald test p-value: Group SFI vs NIT
DataFrame with 6 rows and 6 columns
                 baseMean log2FoldChange     lfcSE      stat      pvalue
                <numeric>      <numeric> <numeric> <numeric>   <numeric>
ENSG00000260187   2.29811       -4.72979  1.411529  -3.35083 8.05698e-04
ENSG00000213816   2.28540       -3.70789  1.101150  -3.36729 7.59117e-04
ENSG00000248332   2.31410       -3.43105  1.113138  -3.08233 2.05390e-03
ENSG00000166523 443.85432       -2.70533  0.591306  -4.57518 4.75816e-06
ENSG00000227158   9.36838       -2.50014  0.669486  -3.73442 1.88150e-04
ENSG00000115602 702.68132       -2.41210  0.647150  -3.72726 1.93570e-04
                      padj
                 <numeric>
ENSG00000260187 0.03569631
ENSG00000213816 0.03460117
ENSG00000248332 0.07075138
ENSG00000166523 0.00084719
ENSG00000227158 0.01316954
ENSG00000115602 0.01341705
```

A continuación se muestran los 5 genes más sobre-regulados.

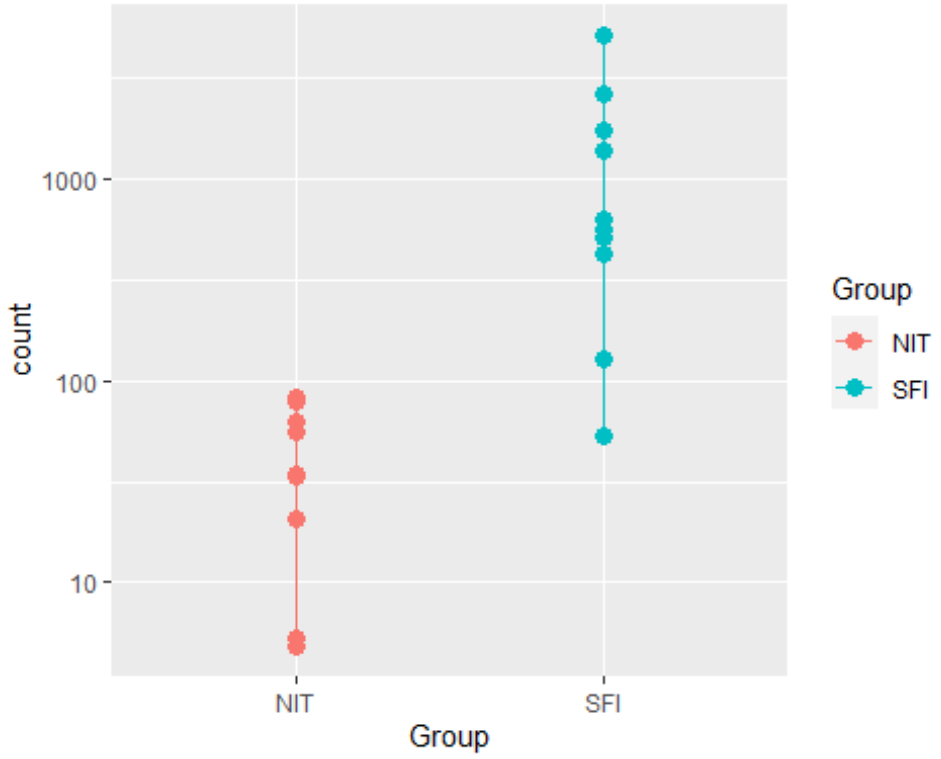
```
head(resSig[ order(resSig$log2FoldChange, decreasing = TRUE), ])
```

```
log2 fold change (MLE): Group SFI vs NIT
Wald test p-value: Group SFI vs NIT
DataFrame with 6 rows and 6 columns
                 baseMean log2FoldChange     lfcSE      stat      pvalue
                <numeric>      <numeric> <numeric> <numeric>   <numeric>
ENSG00000211658   39.5457        7.48723   1.42905   5.23932 1.61173e-07
ENSG00000211667   26.9641        7.25155   1.23200   5.88599 3.95683e-09
ENSG00000211676   30.6060        6.84080   1.28292   5.33222 9.70167e-08
ENSG00000211942  173.6381        6.83028   1.50084   4.55098 5.33976e-06
ENSG00000211611  147.5749        6.71913   1.26928   5.29365 1.19900e-07
ENSG00000235896   44.0506        6.68041   1.83305   3.64442 2.67996e-04
                      padj
                 <numeric>
ENSG00000211658 6.38287e-05
ENSG00000211667 4.01881e-06
ENSG00000211676 4.31846e-05
ENSG00000211942 9.44839e-04
ENSG00000211611 5.09806e-05
ENSG00000235896 1.68536e-02
```

Con estos resultados se puede graficar el conteo de los diversos genes entre los distintos grupos. Por ejemplo, se seleccionó el gen con menor p-valor ajustado: ENSG00000211677 y se puede observar que este presenta más conteos en el grupo SFI.

Se debe recordar que por la fórmula de los resultados, SFI es asumido como el denominador en el Fold Change, por lo cual un mayor conteo en este daría un valor negativo, como se puede confirmar al final del script.

```
topGene <- rownames(res)[which.min(res$padj)]

geneCounts <- plotCounts(ddsdeseq, gene = topGene, intgroup ="Group", ret
urnData = TRUE)

ggplot(geneCounts, aes(x = Group, y = count, color = Group, group = Group
)) +
  scale_y_log10() + geom_point(size = 3) + geom_line()
```




```
#Log2FoldChange
res[rownames(res)=="ENSG00000211677","log2FoldChange"]
```

```
[1] 4.860312
```

MA-Plot

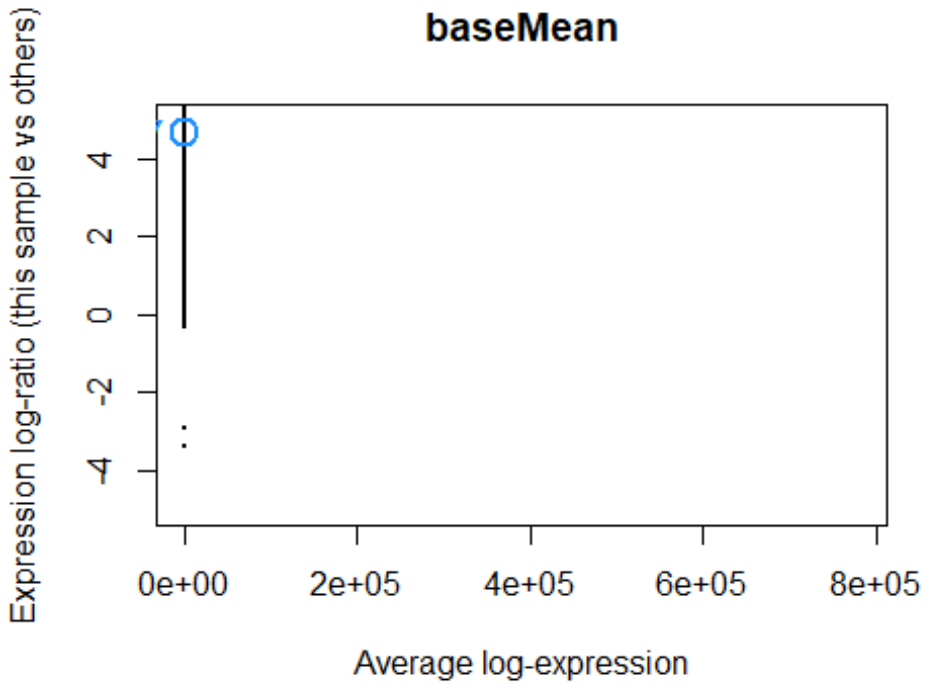
Por otra parte, se realizó un MA-Plot para observar los coeficientes estimados en el modelo, para esto se utilizó la librería apeglm que contrae el log2FoldChange. Así mismo se señala el gen con el menor p-valor ajustado.

El resultado indica que en términos generales hay una sobre-regulación de la muestra SFI en comparación con la NIT.

```
library("apeglm")
resultsNames(ddsdeseq)
```

```
[1] "Intercept"        "Group_SFI_vs_NIT"
```

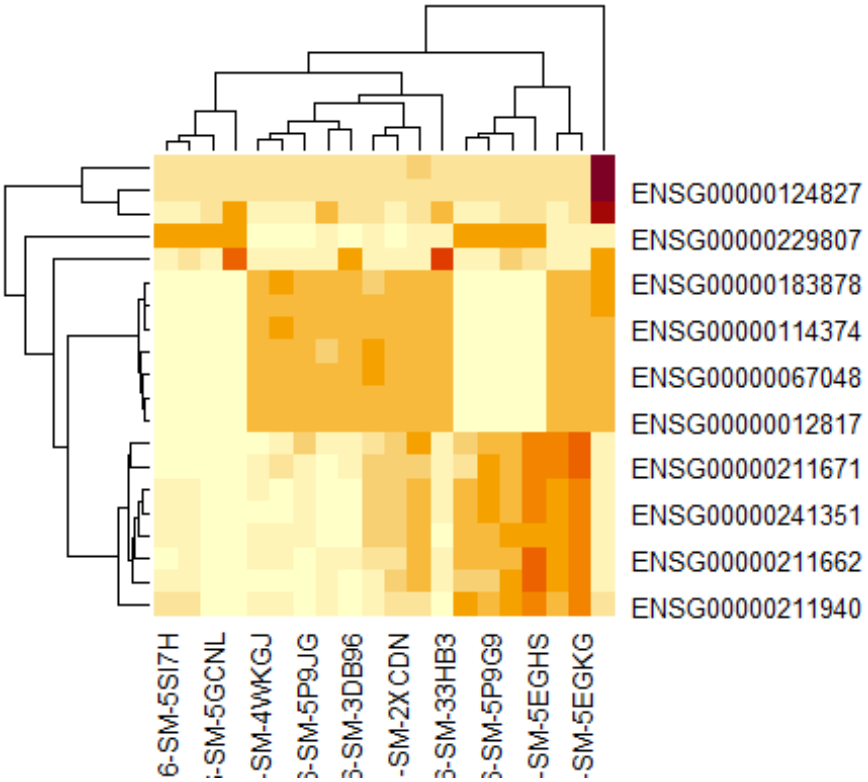
```
res <- lfcShrink(ddsdeseq, coef="Group_SFI_vs_NIT", type="apeglm")

plotMA(res, ylim = c(-5,5))
topGene <- rownames(res)[which.min(res$padj)]
with(res[topGene, ], {
  points(baseMean, log2FoldChange, col="dodgerblue", cex=2, lwd=2)
  text(baseMean, log2FoldChange, topGene, pos=2, col="dodgerblue")
})
```



Clúster de genes

A continuación se realizó un análisis de clúster de los 20 genes con mayor variabilidad, usando los datos con la varianza estabilizada con rlog. Esto permite evidenciar los genes que están más relacionados, lo cual es especialmente útil para la interpretación biológica del análisis.

```
library("genefilter")
topVarGenes <- head(order(rowVars(assay(rld)), decreasing = TRUE), 20)

mat  <- assay(rld)[topVarGenes, ]
mat  <- mat - rowMeans(mat)
anno <- as.data.frame(colData(rld)[, c("Group")])
heatmap(mat, annotation_col = anno)
```



Anotación de los resultados

Continuando con los insumos para la interpretación biológica de este estudio, se procedió a hacer las anotaciones de los genes con los códigos ENSEMBL, para agregar el nombre del gen y el ENTREZID. Para esto se usó la función mapIds para agregar columnas con estos datos.

```
library("AnnotationDbi")
columns(org.Hs.eg.db)
```

```
 [1] "ACCNUM"       "ALIAS"        "ENSEMBL"      "ENSEMBLPROT"  "ENSEMBL
TRANS"
 [6] "ENTREZID"     "ENZYME"       "EVIDENCE"     "EVIDENCEALL"  "GENENAM
E"
[11] "GO"           "GOALL"        "IPI"          "MAP"          "OMIM"
[16] "ONTOLOGY"     "ONTOLOGYALL"  "PATH"         "PFAM"         "PMID"
[21] "PROSITE"      "REFSEQ"       "SYMBOL"       "UCSCKG"       "UNIGENE
"
[26] "UNIPROT"
```

```
library("AnnotationDbi")
row.names(res)=gsub("\\..*","",row.names(res))
res$symbol = mapIds(org.Hs.eg.db,
                    keys=row.names(res),
                    column="SYMBOL",
                    keytype="ENSEMBL",
                    multiVals="first")
```

```
res$entrez = mapIds(org.Hs.eg.db,
                    keys=row.names(res),
                    column="ENTREZID",
                    keytype="ENSEMBL",
                    multiVals="first")

resOrdered <- res[order(res$pvalue),]
head(resOrdered)
```

```
log2 fold change (MAP): Group SFI vs NIT
Wald test p-value: Group SFI vs NIT
DataFrame with 6 rows and 7 columns
                 baseMean log2FoldChange     lfcSE      pvalue        pad
j
                <numeric>      <numeric> <numeric>   <numeric>   <numeric
>
ENSG00000211677  686.0158        4.70812  0.614809 8.61854e-16 2.45525e-1
1
ENSG00000211644  931.5734        4.67870  0.665549 8.94612e-14 1.27429e-0
9
ENSG00000132704  107.9850        5.06425  0.752305 5.74769e-13 5.45801e-0
9
ENSG00000160856  204.3136        4.32717  0.647869 1.04719e-12 7.45809e-0
9
ENSG00000211890 1899.5623        4.25628  0.654212 3.27402e-12 1.86541e-0
8
ENSG00000128438   95.0831        6.06546  0.942086 5.70371e-12 2.70812e-0
8
                     symbol      entrez
                <character> <character>
ENSG00000211677          NA          NA
ENSG00000211644          NA          NA
ENSG00000132704       FCRL2       79368
ENSG00000160856       FCRL3      115352
ENSG00000211890          NA          NA
ENSG00000128438          NA          NA
```

Exportación de resultados

Los resultados fueron finalmente exportados y se colocan a disposición del público en GitHub.

```
resOrderedDF <- as.data.frame(resOrdered)
write.csv(resOrderedDF, file = "results_sfi_nit.csv")
```

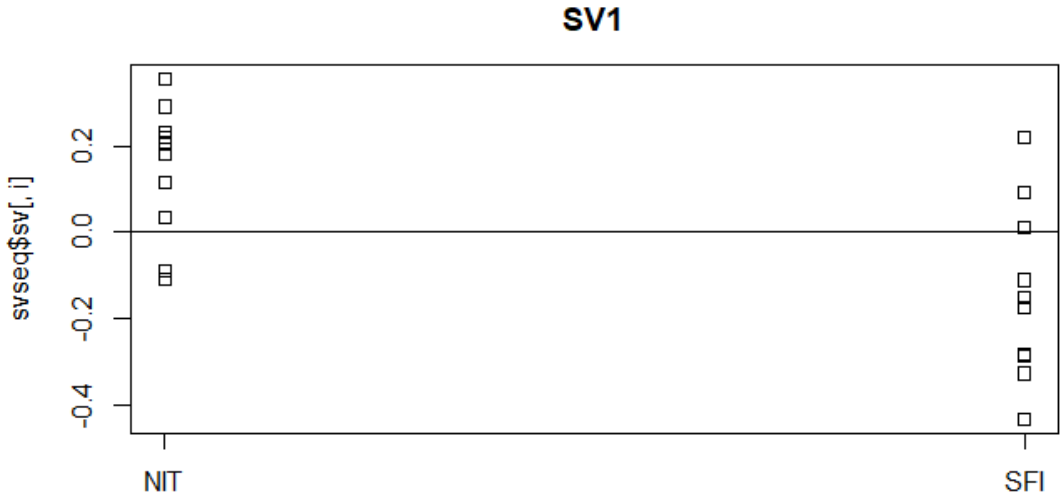
Remoción de efectos ocultos por lotes o batch effects

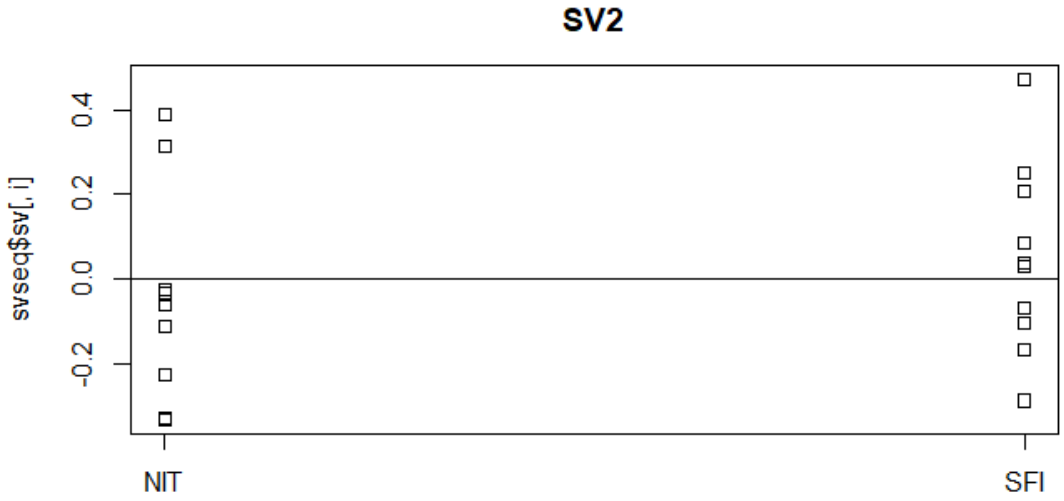
Finalmente se procedió a remover los Batch Effects con la librería SVA, cuya función svaseq permite iterar mínimos cuadrados ponderados para estimar variables

sustitutas, con lo cual se estima la probabilidad de que ser un control, de este modo se detecta la varianza indeseada.

Concretamente se usaron 5 iteraciones. Tras este proceso se detectaron las fuentes ocultas de variación entre grupos.

```
library("sva")

dat  <- counts(ddsdeseq, normalized = TRUE)
idx  <- rowMeans(dat) > 1
dat  <- dat[idx, ]
mod  <- model.matrix(~ Group, colData(ddsdeseq))
mod0 <- model.matrix(~   1, colData(ddsdeseq))
svseq <- svaseq(dat, mod, mod0, n.sv = 2)

Number of significant surrogate variables is:  2
Iteration (out of 5 ):1  2  3  4  5

svseq$sv

             [,1]        [,2]
 [1,] -0.43309084  0.03200602
 [2,] -0.28256674 -0.28466574
 [3,]  0.09249710  0.47181928
 [4,]  0.01267157 -0.16662841
 [5,] -0.11011428  0.20668189
 [6,] -0.32622899 -0.06761484
 [7,] -0.28631568  0.08605505
 [8,] -0.15140336  0.03887925
 [9,] -0.17425478  0.24983906
[10,]  0.21987484 -0.10290923
[11,] -0.08814551 -0.22395362
[12,]  0.11545764 -0.05928223
[13,]  0.29088868 -0.10928101
[14,]  0.20605175 -0.02338128
[15,]  0.03480103  0.31481825
[16,]  0.22033006 -0.03362167
[17,]  0.23121116 -0.33101737
[18,] -0.10733638 -0.06033369
[19,]  0.18028692 -0.32701719
[20,]  0.35538583  0.38960748

par(mfrow = c(2, 1), mar = c(3,5,3,1))
for (i in 1:2) {
  stripchart(svseq$sv[, i] ~ ddsdeseq$Group, vertical = TRUE, main = past
e0("SV", i))
  abline(h = 0)
}
```





Finalmente los datos se utilizan para remover estos efectos en los datos, creando el objeto ddssva, el cual podrá ser utilizado para próximos análisis y estudios.

```
ddssva <- dds
ddssva$SV1 <- svseq$sv[,1]
ddssva$SV2 <- svseq$sv[,2]
design(ddssva) <- ~ SV1 + SV2 + Group
```

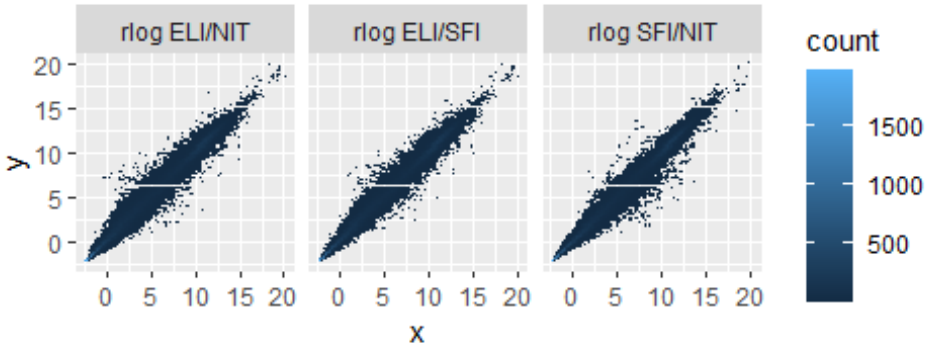
4. Resultados

Tras la detallada explicación de la metodología, a continuación se comparan los resultados de la comparación SFI/NIT, ELI/NIT y ELI/SFI.

Estabilización de la varianza

Como se puede observar, la varianza entre la primera y la segunda muestra de cada comparación es homocedástica, con lo cual se confirma que se realizó correctamente la estabilización de esta.

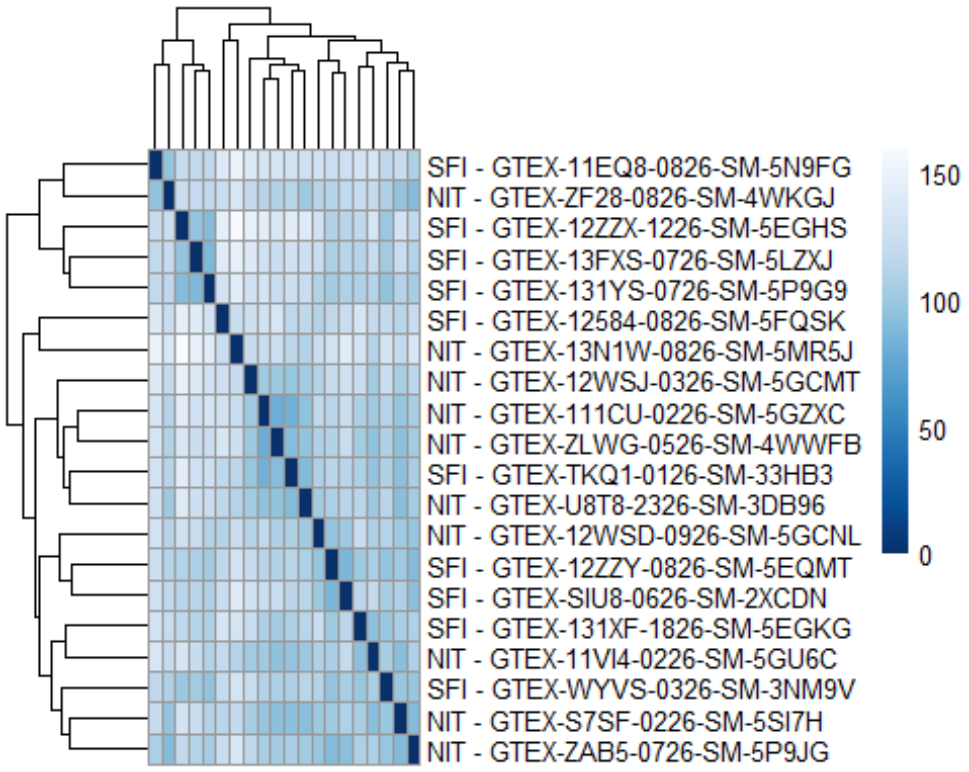
```
ggplot(rbind(df.SFI_NIT,df.ELI_NIT,df.ELI_SFI), aes(x = x, y = y)) + geom
_hex(bins = 80) +
  coord_fixed() + facet_grid( . ~ transformation)
```



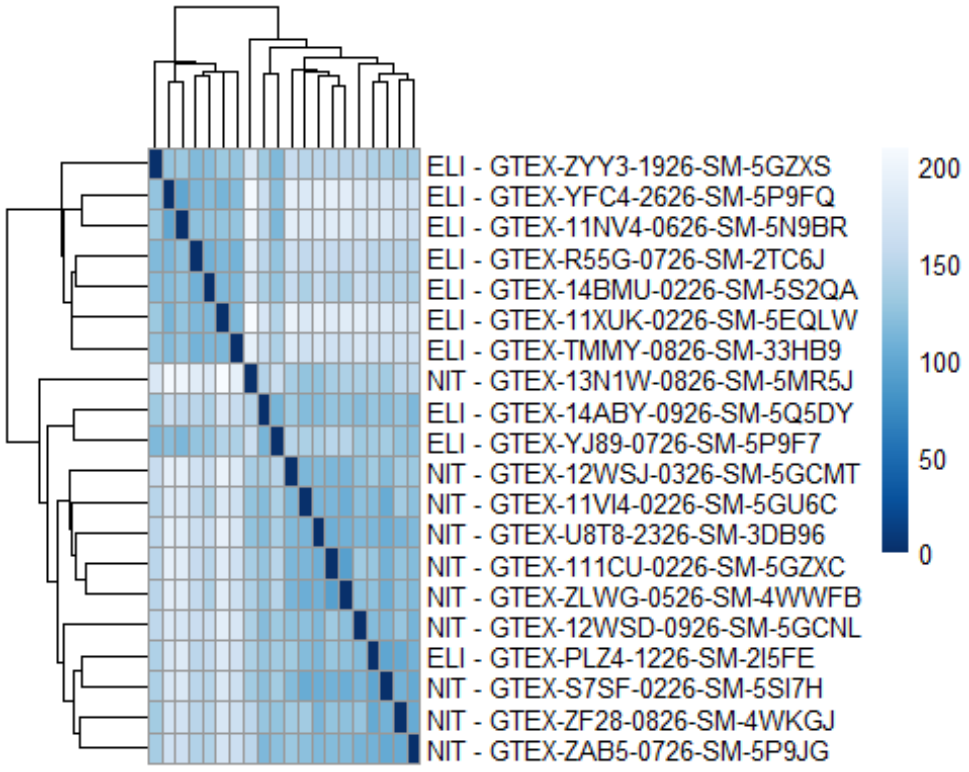
Similitud entre muestras

A continuación se muestran los pheatmap de las comparaciones SFI/NIT, ELI/NIT y ELI/SFI. Los resultados demuestran las muestras más parecidas en un azul más oscuro y con el dendograma se evidencian las aglomeraciones de estas. Como cabría esperar, las muestras de un mismo grupo se presentan más parecidas en términos generales, aunque se requiere un análisis a mayor profundidad sobre el significado biológico de esto.

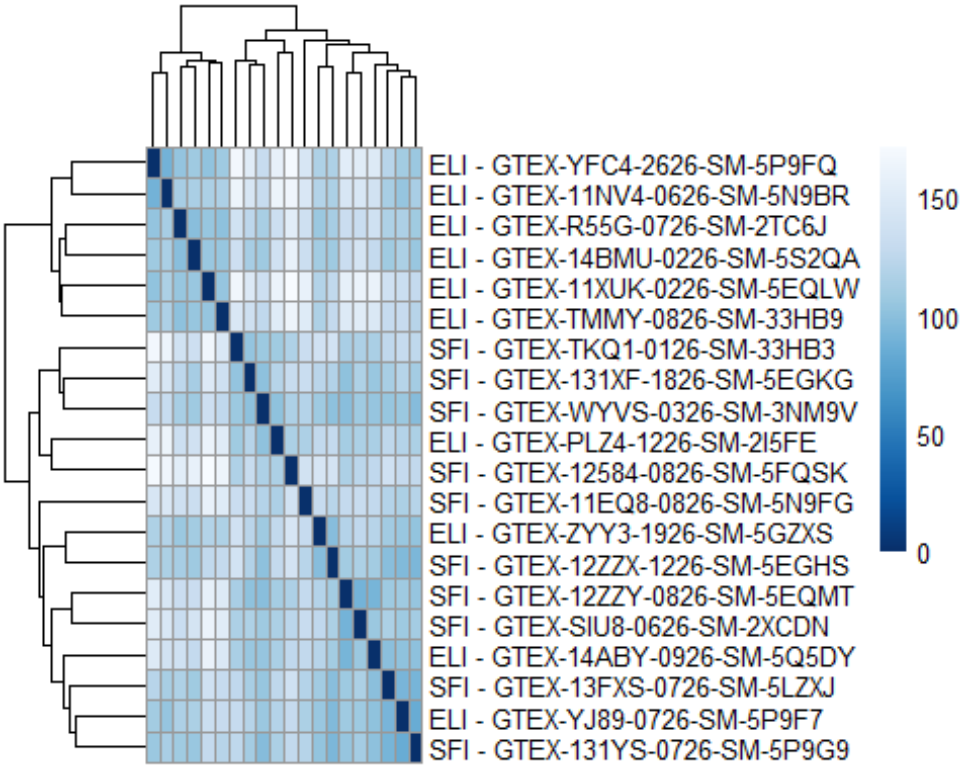
```
ph.graph.SFI_NIT
```




```
ph.graph.ELI_NIT
```




```
ph.graph.ELI_SFI
```

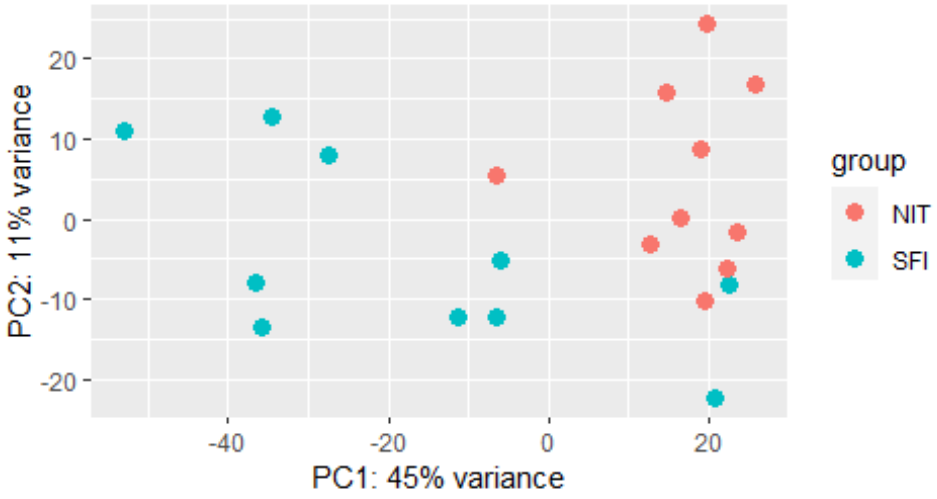


PCA-Plot

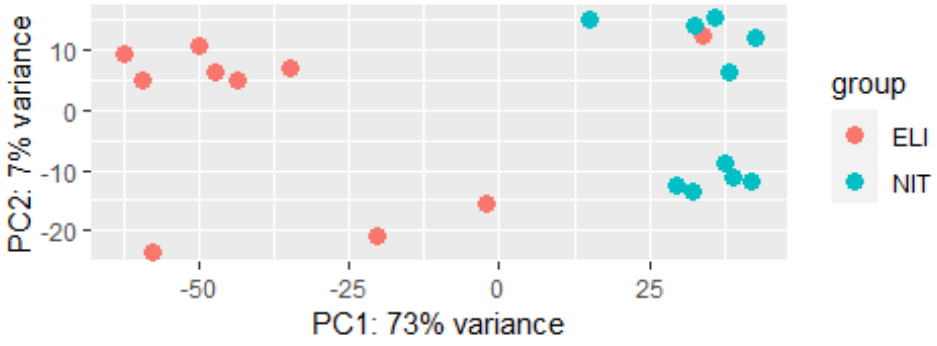
Los diagramas de dispersión entre la primera y segunda componente del PCA demuestran que en términos generales los grupos se pueden distinguir casi perfectamente, por lo cual se observa que es viable realizar un análisis de expresión diferencial.

Se resalta que la comparción ELI/NIT fue la que presentó una mayor diferenciación, mientras que ELI/SFI presentó un mayor número de muestras cercanas, lo cual puede ser lógico por el aparente parentesco entre ambas patologías.

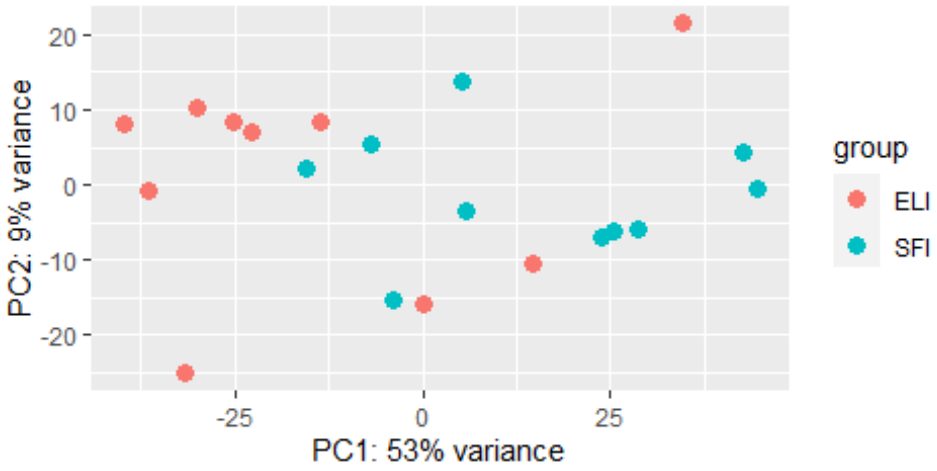
```
plotPCA(rld.SFI_NIT, intgroup = c("Group"))
```




```
plotPCA(rld.ELI_NIT, intgroup = c("Group"))
```



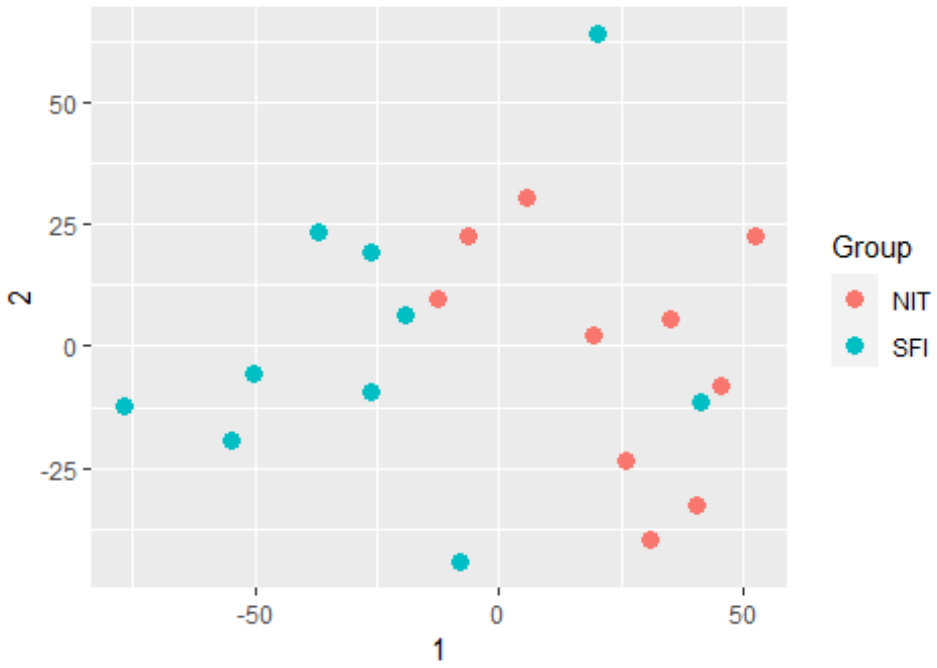

```
plotPCA(rld.ELI_SFI, intgroup = c("Group"))
```



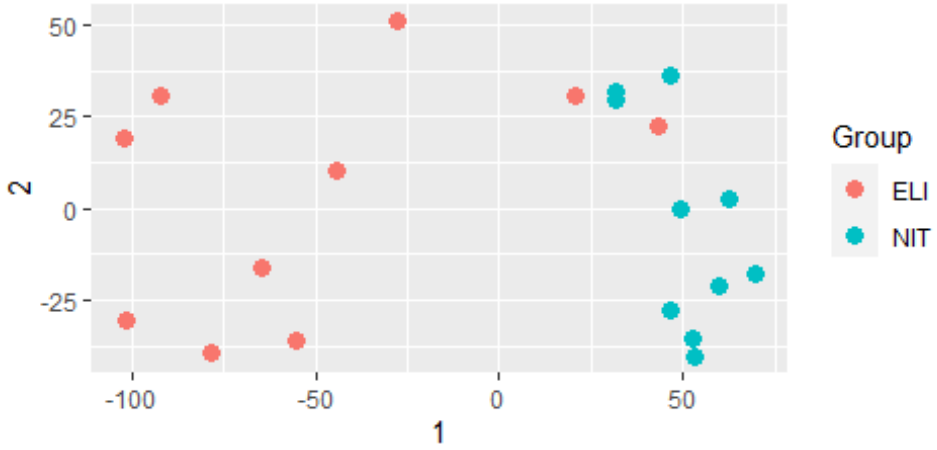
MDS-Plot

En cuanto a los gráficos del MDS, los resultados confirman el análisis realizado con el PCA. Ya que las muestras presentan diferencias evidentes, sin embargo la comparación ELI/SFI fue la que menos diferenciación presenta.

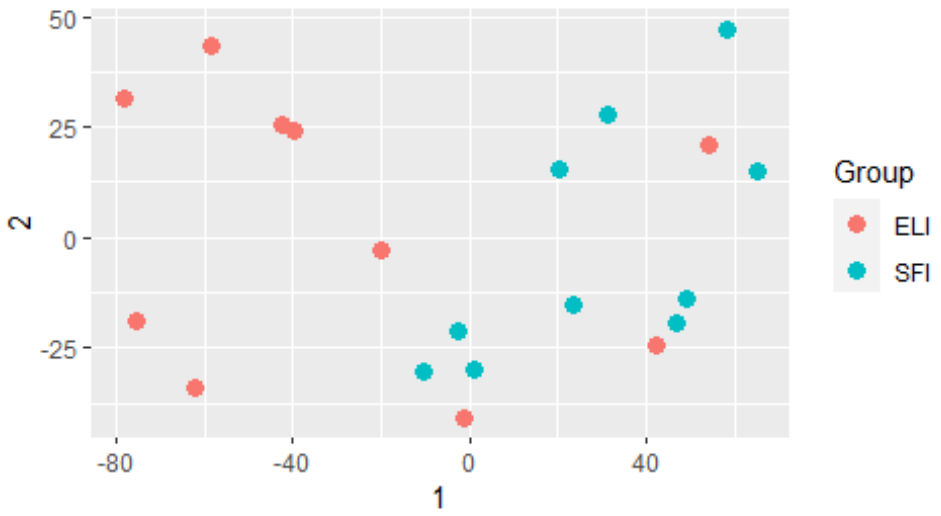
```
ggplot(mds, aes(x = `1`, y = `2`, color = Group)) +
  geom_point(size = 3) + coord_fixed()
```




```
ggplot(mds.ELI_NIT, aes(x = `1`, y = `2`, color = Group)) +
  geom_point(size = 3) + coord_fixed()
```



```
ggplot(mds.ELI_SFI, aes(x = `1`, y = `2`, color = Group)) +
  geom_point(size = 3) + coord_fixed()
```



Expresión diferencial

Los resultados demuestran que 834 (2%) de los genes SFI están sobre regulados en comparación con los de las muestras NIT, así mismo 166 (0.4%) de estos están sub-regulados.

En cambio, las mayores diferencias se obseran entre ELI y NIT, como se pudo apreciar en los gráficos, y es que 4105 (9.8%) genes ELI están sobre regulados mientras que 1941 (4.6%) están subregulados.

Finalmente, también existen grandes diferencias entre ELI y SFI, ya que 1784 (4.2%) genes están sobre regulados y 575 (1.4%) están sub regulados.

```
[1] "SFI_NIT"


out of 41299 with nonzero total read count
adjusted p-value < 0.1
LFC > 0 (up)       : 834, 2%
LFC < 0 (down)     : 166, 0.4%
outliers [1]       : 0, 0%
low counts [2]     : 12814, 31%
(mean count < 2)
```

```
[1] see 'cooksCutoff' argument of ?results
[2] see 'independentFiltering' argument of ?results
```

```
[1] "ELI_NIT"
```

```


out of 41778 with nonzero total read count
adjusted p-value < 0.1
LFC > 0 (up)       : 4105, 9.8%
LFC < 0 (down)     : 1941, 4.6%
outliers [1]       : 0, 0%
low counts [2]     : 10538, 25%
(mean count < 1)
[1] see 'cooksCutoff' argument of ?results
[2] see 'independentFiltering' argument of ?results
```

```
[1] "ELI_SFI"
```

```


out of 42226 with nonzero total read count
adjusted p-value < 0.1
LFC > 0 (up)       : 1784, 4.2%
LFC < 0 (down)     : 575, 1.4%
outliers [1]       : 0, 0%
low counts [2]     : 12286, 29%
(mean count < 1)
[1] see 'cooksCutoff' argument of ?results
[2] see 'independentFiltering' argument of ?results
```

Cabe destacar que el análisis biológico sobre el significado de estos genes y estas diferencias, está por fuera del alcance de este estudio, ya que el proposito es primordialmente técnico.

Plot-Counts

A continuación se muestra el plot count para el gen con menor p-valor ajustado de cada comparación.

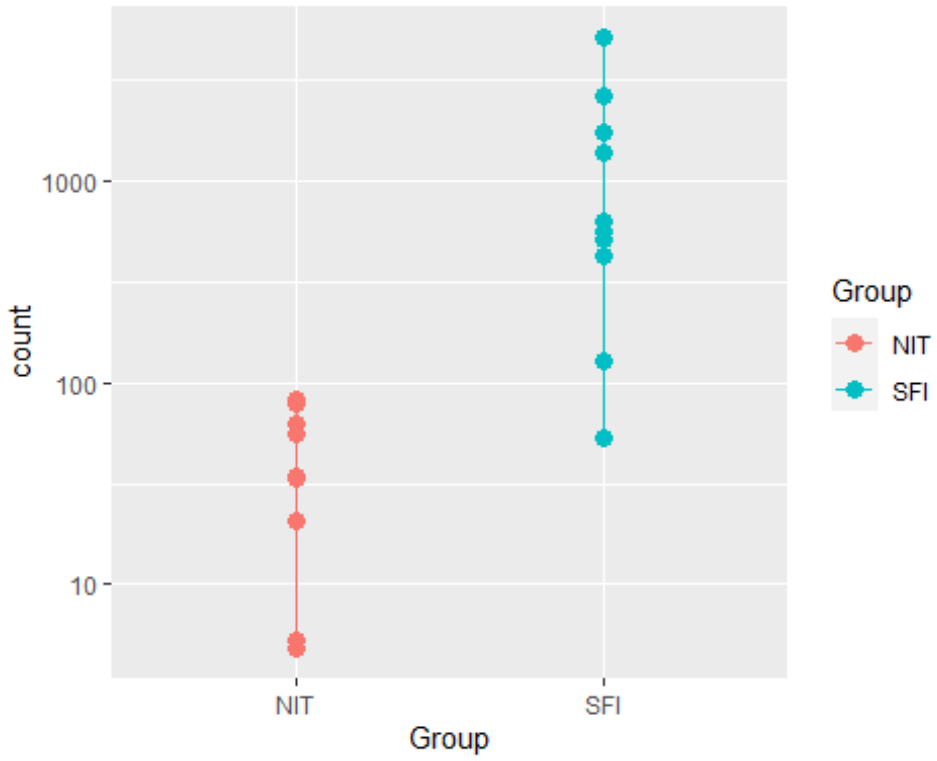
En el primer caso (SFI/NIT) se observa que el conteo es mayor en en SFI, por lo cual este gen está sobre-regulado. En el segundo (ELI/SFI), se evidencia que el conteo es superior en ELI, por lo cual está sub-regulado. Finalmente, en el último caso (ELI/SFI) también el conteo es superior en ELI por lo cual está sobre regulado.

Cabe destacar que estas comparaciones son realizadas con genes diferentes, por lo cual no se puede analizar los tres gráficos en conjunto.

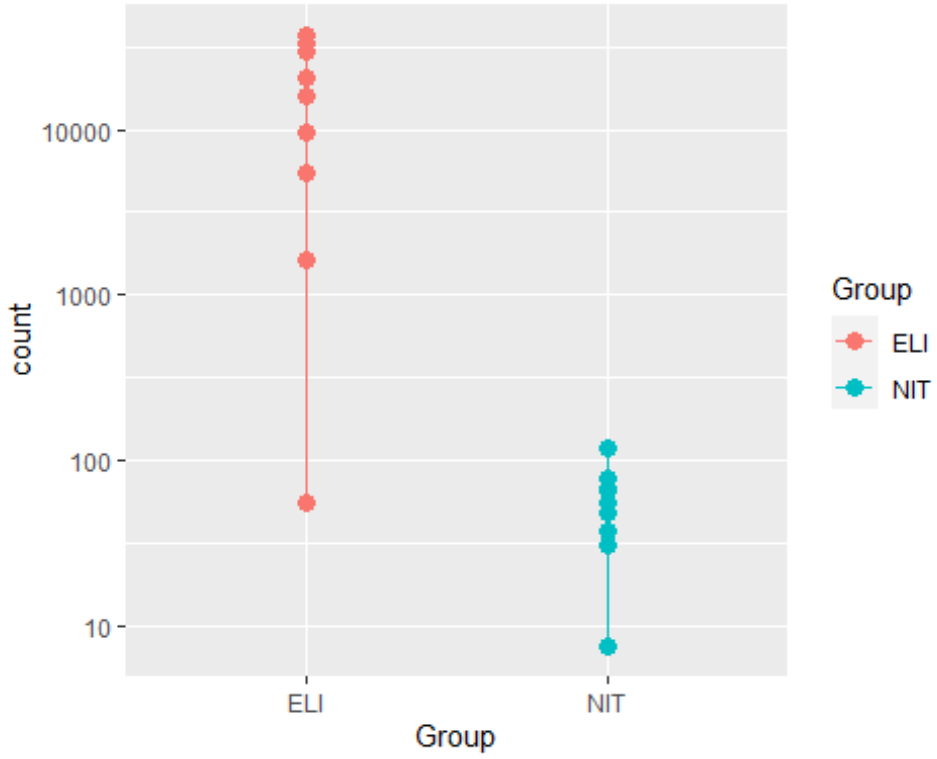
```
topGene
```

```
[1] "ENSG00000211677"
```

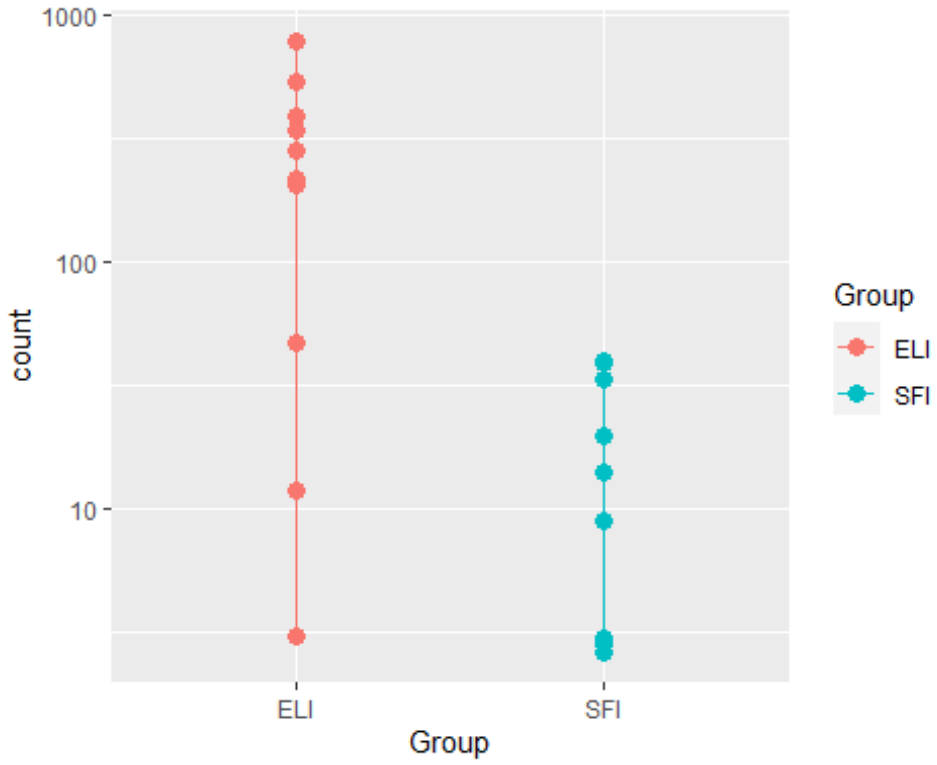
```
pc.SFI_NIT
```




```
topGene.ELI_NIT

[1] "ENSG00000156738"

pc.ELI_NIT
```




```
topGene.ELI_SFI

[1] "ENSG00000197520"

pc.ELI_SFI
```

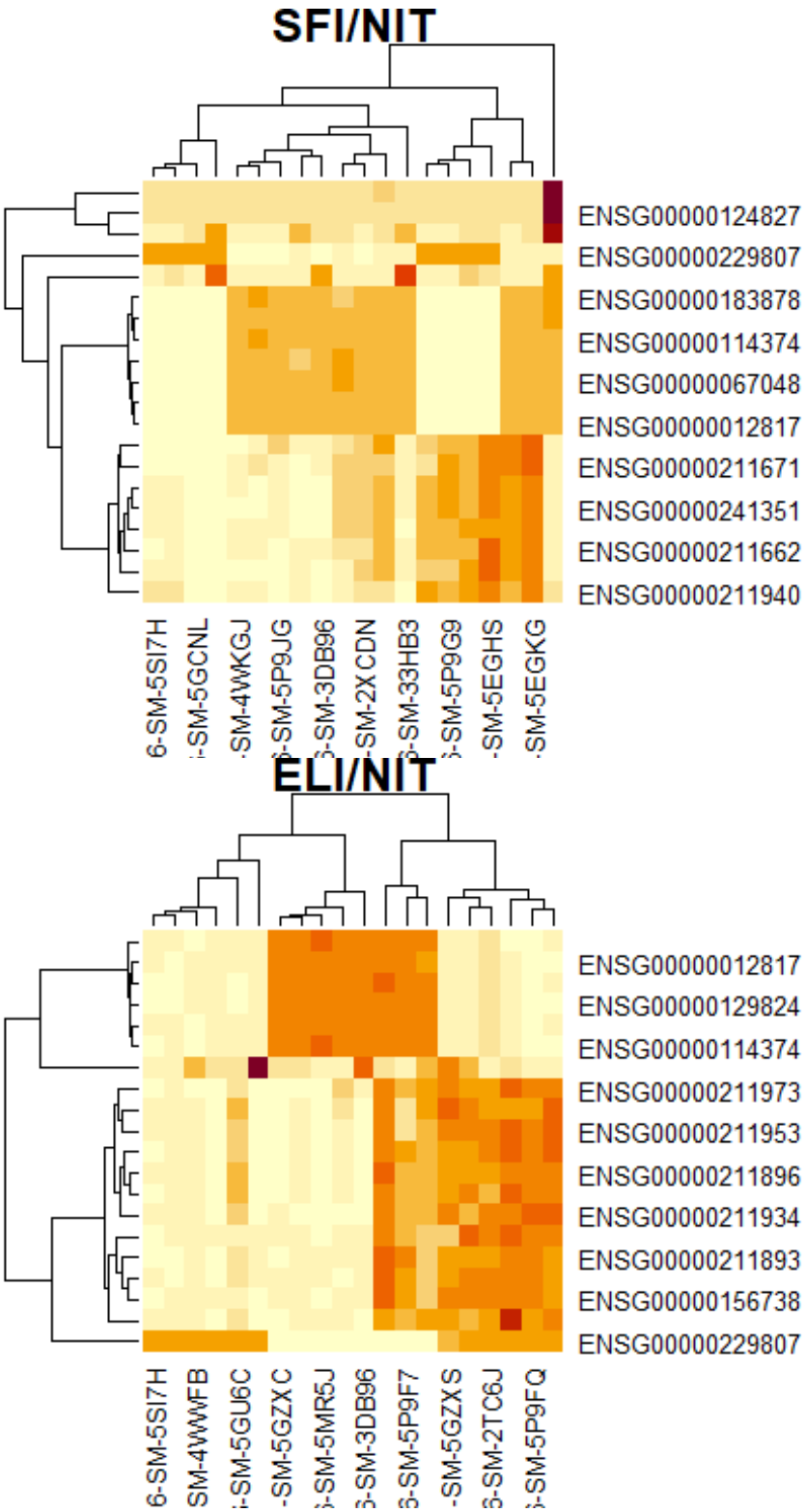


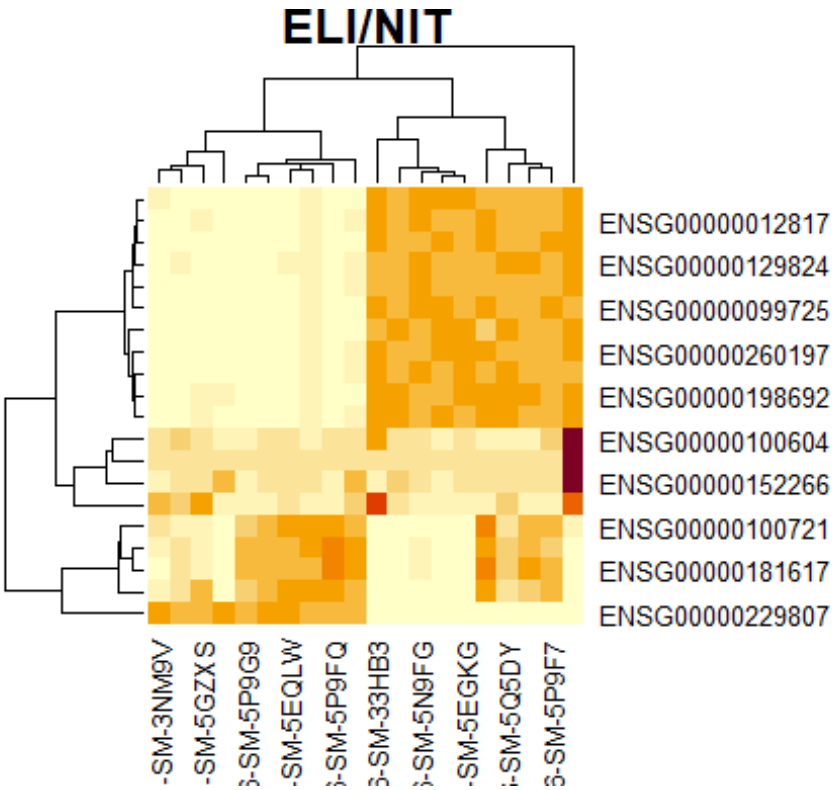
MA - Plot

En cuanto al MA-Plot este no ofreció mayor información sobre los distintos grupos. Por esta razón se prescinde de su utilización.

Clúster de Genes

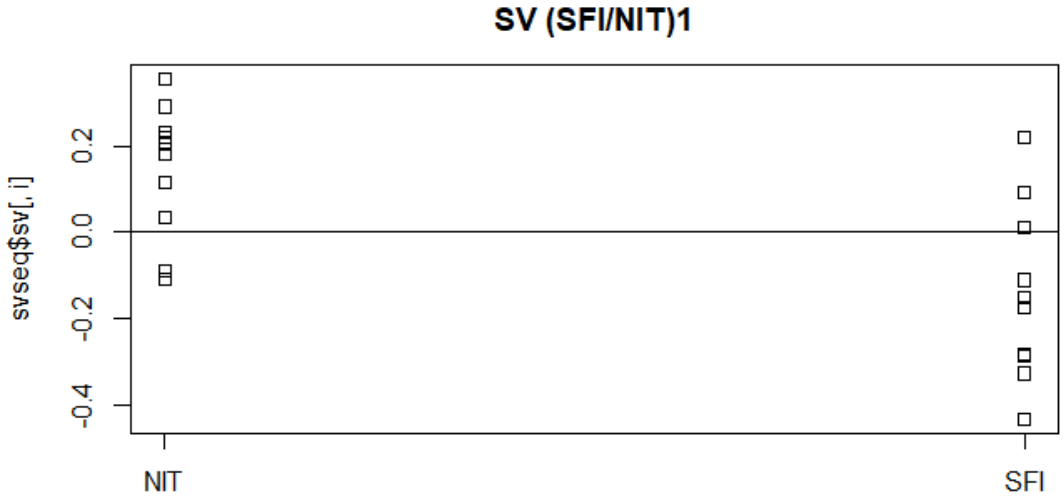
A continuación se muestra un análisis de clúster de los genes con menor p-valor, es decir de aquellos cuya diferencia fue más estadísticamente significativa. Cabe destacar que esta presentación no se analiza a mayor profundidad debido a que corresponde a un equipo de biólogos y genetistas realizar la evaluación de la significancia biológica de estos datos.

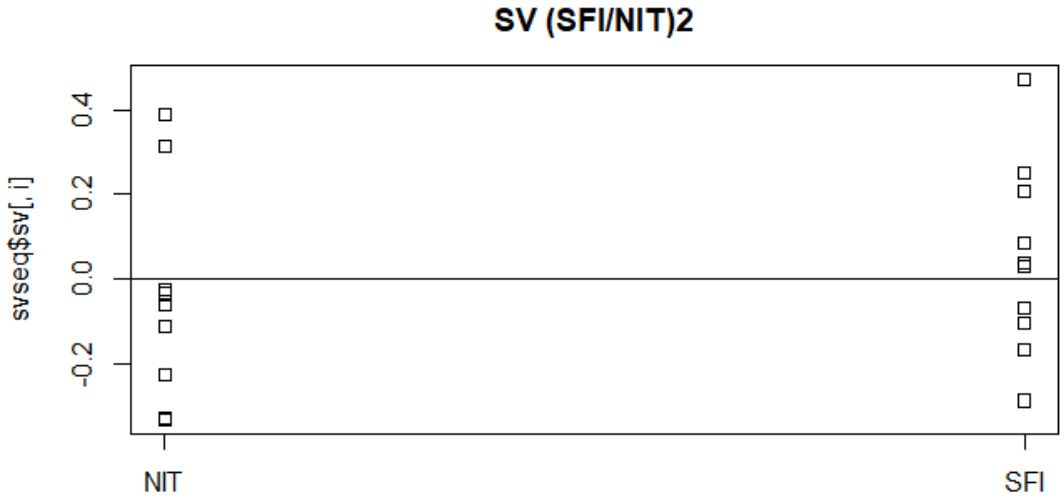

SFI/NIT
ENSG00000124827
ENSG00000229807
ENSG00000183878
ENSG00000114374
ENSG00000067048
ENSG00000012817
ENSG00000211671
ENSG00000241351
ENSG00000211662
ENSG00000211940
6-SM-5SI7H
-SM-5GCNL
-SM-4WKGJ
-SM-5P9JG
-SM-3DB96
-SM-2XCDN
-SM-33HB3
-SM-5P9G9
-SM-5EGHS
-SM-5EGKG
ELI/NIT
ENSG00000012817
ENSG00000129824
ENSG00000114374
ENSG00000211973
ENSG00000211953
ENSG00000211896
ENSG00000211934
ENSG00000211893
ENSG00000156738
ENSG00000229807
6-SM-5SI7H
SM-4WWFB
-SM-5GU6C
-SM-5GZXC
-SM-5MR5J
-SM-3DB96
-SM-5P9F7
-SM-5GZXS
-SM-2TC6J
-SM-5P9FQ

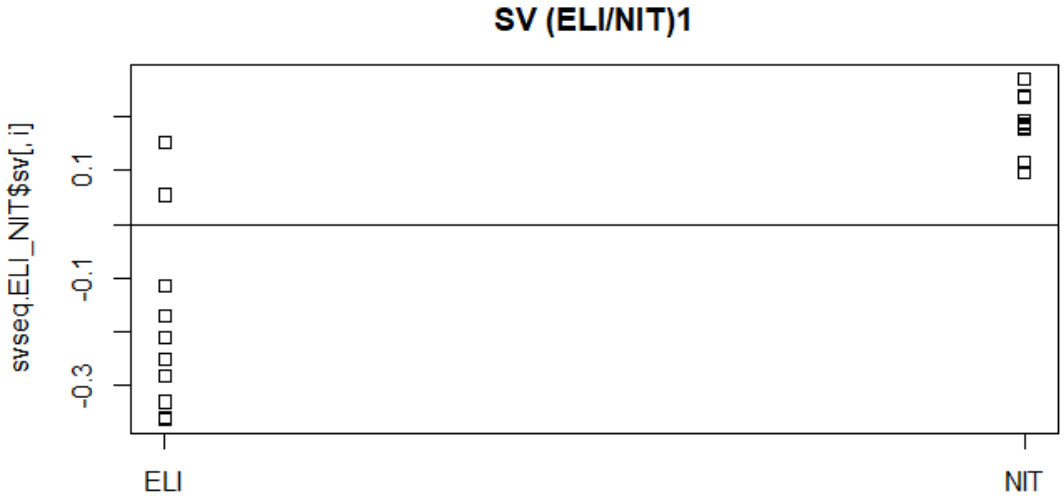


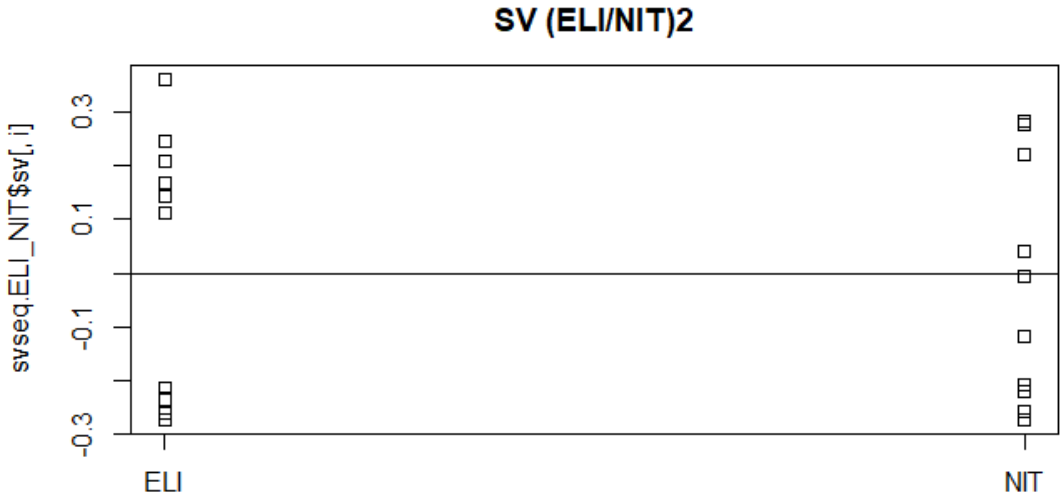
Remoción de Batch Effects

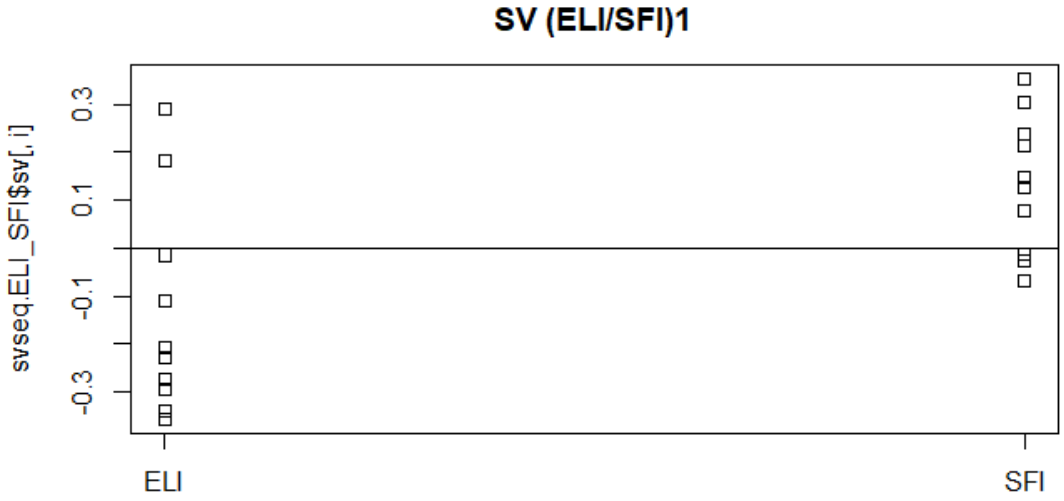
Para finalizar, se hace una remoción de los Batch Effects de las tres muestras, con lo cual se podrá contar con bases con una detección de las fuentes de variación ocultas.

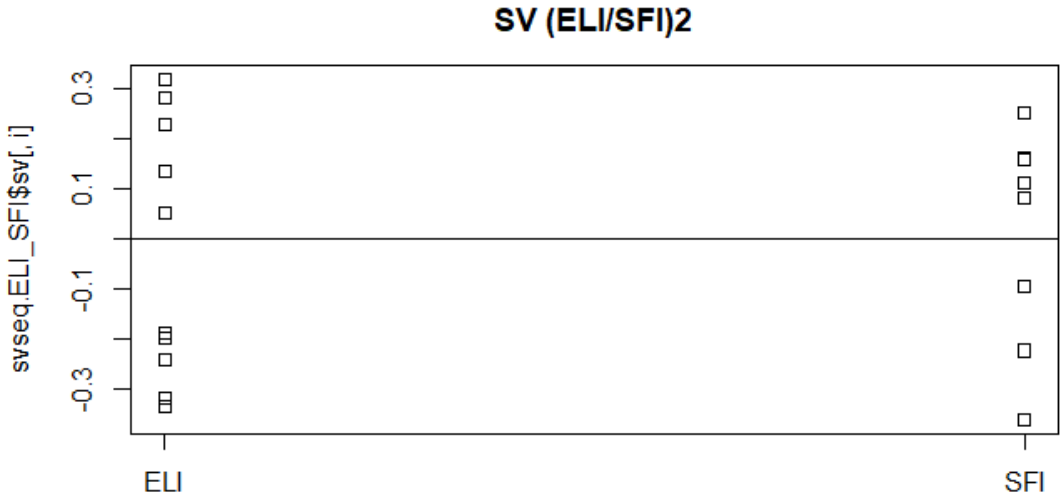

SV (SFI/NIT)1
svseq$sv[, i]
0.2
0.0
-0.2
-0.4
NIT
SFI


SV (SFI/NIT)2
svseq$sv[, i]
0.4
0.2
0.0
-0.2
NIT
SFI


SV (ELI/NIT)1
svseq.ELI_NIT$sv[, i]
0.1
-0.1
-0.3
ELI
NIT


SV (ELI/NIT)2
svseq.ELI_NIT$sv[, i]
0.3
0.1
-0.1
-0.3
ELI
NIT





Anotación y exportación de resultados

```
[1] ".SFI_NIT"

log2 fold change (MAP): Group SFI vs NIT
Wald test p-value: Group SFI vs NIT
DataFrame with 6 rows and 7 columns
                 baseMean log2FoldChange     lfcSE      pvalue        pad
j
                <numeric>      <numeric> <numeric>   <numeric>   <numeric
>
ENSG00000211677  686.0158        4.70812  0.614809 8.61854e-16 2.45525e-1
1
ENSG00000211644  931.5734        4.67870  0.665549 8.94612e-14 1.27429e-0
9
ENSG00000132704  107.9850        5.06425  0.752305 5.74769e-13 5.45801e-0
9
```

```
ENSG00000160856  204.3136        4.32717  0.647869 1.04719e-12 7.45809e-0
9
ENSG00000211890 1899.5623        4.25628  0.654212 3.27402e-12 1.86541e-0
8
ENSG00000128438   95.0831        6.06546  0.942086 5.70371e-12 2.70812e-0
8
                     symbol      entrez
                <character> <character>
ENSG00000211677          NA          NA
ENSG00000211644          NA          NA
ENSG00000132704       FCRL2       79368
ENSG00000160856       FCRL3      115352
ENSG00000211890          NA          NA
ENSG00000128438          NA          NA

[1] ".ELI_NIT"

log2 fold change (MLE): Group ELI vs NIT
Wald test p-value: Group ELI vs NIT
DataFrame with 6 rows and 8 columns
                 baseMean log2FoldChange     lfcSE      stat      pvalue
                <numeric>      <numeric> <numeric> <numeric>   <numeric>
ENSG00000156738   8494.96        8.25309  0.581412   14.1949 9.85265e-46
ENSG00000163534   1481.67        7.91858  0.606503   13.0561 5.86445e-39
ENSG00000167483   1310.53        7.37686  0.577200   12.7804 2.10895e-37
ENSG00000132704   1060.29        8.65261  0.680182   12.7210 4.51919e-37
ENSG00000128438    655.49        9.17441  0.723544   12.6798 7.65077e-37
ENSG00000177455   1821.27        8.35424  0.665988   12.5441 4.28136e-36
                      padj      symbol      entrez
                 <numeric> <character> <character>
ENSG00000156738 3.07876e-41       MS4A1         931
ENSG00000163534 9.16261e-35       FCRL1      115350
ENSG00000167483 2.19669e-33      NIBAN3      199786
ENSG00000132704 3.53039e-33       FCRL2       79368
ENSG00000128438 4.78143e-33          NA          NA
ENSG00000177455 2.22973e-32        CD19         930

[1] ".ELI_SFI"

log2 fold change (MLE): Group ELI vs SFI
Wald test p-value: Group ELI vs SFI
DataFrame with 6 rows and 8 columns
                 baseMean log2FoldChange     lfcSE      stat      pvalue
                <numeric>      <numeric> <numeric> <numeric>   <numeric>
ENSG00000161929  389.8446        3.00906  0.479412   6.27657 3.46133e-10
ENSG00000129173   59.6473        3.53186  0.574325   6.14959 7.76833e-10
ENSG00000169679  203.0674        2.73544  0.447025   6.11921 9.40380e-10
ENSG00000197520  148.7412        4.03076  0.659196   6.11466 9.67630e-10
ENSG00000111913 2235.5150        2.53675  0.417063   6.08242 1.18383e-09
ENSG00000160505   28.6687        5.81215  0.977550   5.94563 2.75402e-09
                      padj      symbol      entrez
```

```
                  <numeric> <character> <character>
ENSG00000161929 7.09017e-06       SCIMP      388325
ENSG00000129173 7.09017e-06        E2F8       79733
ENSG00000169679 7.09017e-06        BUB1         699
ENSG00000197520 7.09017e-06     FAM177B      400823
ENSG00000111913 7.09017e-06      RIPOR2        9750
ENSG00000160505 1.37453e-05       NLRP4      147945
```

Discusión

Los resultados de cada comparación fueron exportados en archivos .csv que pueden ser consultados en el repositorio de GitHub de este estudio.

https://github.com/alcastaro/PEC2_OMICAS_ACA

Conclusiones

Se requiere continuar el estudio con un análisis de la significancia biológica de los resultados.