Salvador\_Escribano\_ADO\_PEC2

Pedro Salvador Escribano

4/6/2020

Table of Contents

<https://github.com/pesales/PEC2-An-lisis-de-datos-micos>

# Abstract

Se realizó un análisis de RNA-Seq a partir de los datos proporcionados. Se efectuó una extracción aleatoria de 10 muestras por grupo de un total de 3 grupos de muestras a analizar y se buscaron diferencias por pares. El análisis aportó las listas de genes diferencialmente expresados en cada una de las comparaciones

# Objetivos

• Realizar una extracción aleatoria a partir de unos datos proporcionados.

• Llevar a cabo el análisis de expresión diferencial de los datos extraidos en el primer objetivo.

# Materiales y Métodos

## Naturaleza de los datos, tipo de experimento, diseño experimental

Los datos proporcionados para la realización de esta actividad corresponden a muestras de un estudio obtenido del repositorio (GTEx). 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:

• Not infiltrated tissues (NIT): 236 samples

• Small focal infiltrates (SFI): 42 samples

• Extensive lymphoid infiltrates (ELI): 14 samples.

## Métodos y herramientas que habéis utilizado en el análisis:

El análisis de los datos será llevado a cabo en Rstudio, principalmente con las herramientas aportadas con el paquete DESeq2

### Procedimiento general de análisis (pasos, “workflow” o “pipeline” que habéis seguido)

El proceso de análisis se ha estratificado de la siguiente manera: Definición de los datos. Preprocesado de los datos: filtraje y normalización. Identificación de genes diferencialmente expresados. Anotación de los resultados. Exportación de los resultados.

# Resultados

Se ha obtenido las listas de genes cuya expresión se ve alterada en las diferentes condiciones analizadas y se ha almacenado dicha información en tablas en formato CSV. Para más detalle, consultar apéndice.

# Discusión

En mi opinión, la mayor limitación que ofrece este tipo de estudios es el análisis estadístico de las diferencias en la expresión. Al ser un número de genes muy alto el que se analiza, los métodos estadísticos estandar fallan, reportando una gran cantidad de falsos positivos, por lo que se tiende a la realización de un ajuste del p-valor. La elección del nivel de significatividad adecuado condiciona notablemente los resultados obtenidos. Por otra parte, considero imprescindible una comunicación fluida entre el investigador y el analista bioinformático con el fin de no perder información que pueda ser relevante, pues es posible que, durante el procesado de los datos, información que aparentemente no es significativa para el analísta, sea de gran relevancia para el investigador por tener una relación más estrecha con el experimento.

# Notas

Se ha omitido el (“Gene Enrichment Analysis”) al ser el número de genes que presentan diferencias en su expresión demasiado pequeño como para llevar a cabo este paso.

La anotación de los datos no se ha podido realizar completamente al haber algunos genes incluidos en el análisis que no tienen una entrada válida en la base de datos utilizadas. Para esos casos, se mantiene la nomenclatura inicial para poder ser consultada en caso de ser necesaria.

# Apéndice

## Definición de los datos

### Importación de datos:

counts=read.csv("C:/Users/peri8/Google Drive/Master UOC/Análisis de datos ómicos/PEC2/counts.csv", sep = ";")  
targets=read.csv("C:/Users/peri8/Google Drive/Master UOC/Análisis de datos ómicos/PEC2/targets.csv")

### Selección de muestra con n=10 para cada uno de los grupos:

# Separo los 3 grupos:  
NIT=subset(targets,Group=="NIT")  
SFI=subset(targets,Group=="SFI")  
ELI=subset(targets,Group=="ELI")  
# Obtengo un muestreo aleatorio de 10 muestras por grupo en cada uno de los grupos:  
set.seed(123)  
require(dplyr)  
sampleNIT=sample\_n(NIT,size = 10)  
sampleSFI=sample\_n(SFI,size = 10)  
sampleELI=sample\_n(ELI,size = 10)  
# Uno todos los muestreos en un solo data frame:  
sampletargets=rbind(sampleNIT,sampleSFI,sampleELI)  
# Extraigo las muestras seleccionadas del archivo counts:  
rawCountTable=counts[,c(sampletargets$Sample\_Name)]  
rownames(rawCountTable)=counts$X  
dim(rawCountTable)

## [1] 56202 30

dim(sampletargets)

## [1] 30 9

### Construcción del objeto DESeqDataSet:

dds <- DESeqDataSetFromMatrix(countData = rawCountTable,colData = sampletargets, design = ~ Group)  
dds

## class: DESeqDataSet   
## dim: 56202 30   
## metadata(1): version  
## assays(1): counts  
## rownames(56202): ENSG00000223972.4 ENSG00000227232.4 ...  
## ENSG00000210195.2 ENSG00000210196.2  
## rowData names(0):  
## colnames(30): GTEX.S341.0226.SM.5S2VG GTEX.ZAK1.0726.SM.5HL8Q ...  
## GTEX.13QBU.0626.SM.5J2OG GTEX.YFCO.0326.SM.4W1ZP  
## colData names(9): Experiment SRA\_Sample ... Group ShortName

## Análisis exploratorio y visualización:

### Prefiltrado de los datos:

nrow(dds)

## [1] 56202

dds <- dds[ rowSums(counts(dds)) > 1, ] # Elimino las filas con expresiones despreciables (menores o iguales a 1)  
nrow(dds)

## [1] 43513

### Normalización:

Realizo una normalización de los datos para poder llevar a cabo el análisis. Ya que se requiere homocedasticidad, considero el método vst el más adecuado para esta transformación.

vsd <- vst(dds, blind = FALSE)  
assay(vsd)[1:5, 1:5]

## GTEX.S341.0226.SM.5S2VG GTEX.ZAK1.0726.SM.5HL8Q  
## ENSG00000223972.4 5.008719 4.589095  
## ENSG00000227232.4 9.010059 9.376409  
## ENSG00000243485.2 4.886335 4.898651  
## ENSG00000237613.2 5.008719 5.026041  
## ENSG00000268020.2 4.589095 4.898651  
## GTEX.X15G.0526.SM.3NMB7 GTEX.11DYG.0826.SM.5N9GH  
## ENSG00000223972.4 6.066370 5.098513  
## ENSG00000227232.4 9.821541 9.589787  
## ENSG00000243485.2 5.322253 4.589095  
## ENSG00000237613.2 5.379618 4.589095  
## ENSG00000268020.2 5.189829 4.589095  
## GTEX.Y5V6.0526.SM.4VBRV  
## ENSG00000223972.4 5.054871  
## ENSG00000227232.4 8.824796  
## ENSG00000243485.2 5.054871  
## ENSG00000237613.2 4.969947  
## ENSG00000268020.2 4.969947

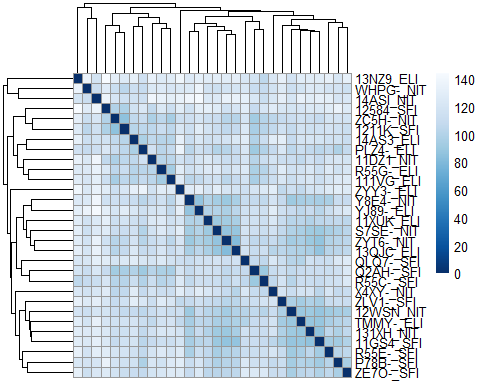
colData(vsd)

## DataFrame with 30 rows and 10 columns  
## Experiment SRA\_Sample Sample\_Name  
## <factor> <factor> <factor>  
## GTEX.S341.0226.SM.5S2VG SRX221539 SRS389105 GTEX-S7SE-0726-SM-2XCD7  
## GTEX.ZAK1.0726.SM.5HL8Q SRX577234 SRS629488 GTEX-ZC5H-0626-SM-5LU9K  
## GTEX.X15G.0526.SM.3NMB7 SRX407742 SRS524447 GTEX-X4XY-0826-SM-4E3JM  
## GTEX.11DYG.0826.SM.5N9GH SRX579805 SRS629854 GTEX-11DZ1-2726-SM-5A5KH  
## GTEX.Y5V6.0526.SM.4VBRV SRX590541 SRS634561 GTEX-Y8E4-0126-SM-4VBQ2  
## ... ... ... ...  
## GTEX.111FC.1026.SM.5GZX1 SRX563960 SRS625636 GTEX-111VG-0526-SM-5N9BW  
## GTEX.TKQ1.0126.SM.33HB3 SRX222429 SRS389623 GTEX-TMMY-0826-SM-33HB9  
## GTEX.ZYW4.1126.SM.5SI99 SRX568364 SRS627095 GTEX-ZYY3-1926-SM-5GZXS  
## GTEX.13QBU.0626.SM.5J2OG SRX601511 SRS638114 GTEX-13QJC-0826-SM-5RQKC  
## GTEX.YFCO.0326.SM.4W1ZP SRX583148 SRS631283 GTEX-YJ89-0726-SM-5P9F7  
## Grupo\_analisis body\_site molecular\_data\_type  
## <integer> <factor> <factor>  
## GTEX.S341.0226.SM.5S2VG 1 Thyroid RNA Seq (NGS)  
## GTEX.ZAK1.0726.SM.5HL8Q 1 Thyroid RNA Seq (NGS)  
## GTEX.X15G.0526.SM.3NMB7 1 Thyroid Allele-Specific Expression  
## GTEX.11DYG.0826.SM.5N9GH 1 Thyroid Allele-Specific Expression  
## GTEX.Y5V6.0526.SM.4VBRV 1 Thyroid RNA Seq (NGS)  
## ... ... ... ...  
## GTEX.111FC.1026.SM.5GZX1 3 Thyroid RNA Seq (NGS)  
## GTEX.TKQ1.0126.SM.33HB3 3 Thyroid Allele-Specific Expression  
## GTEX.ZYW4.1126.SM.5SI99 3 Thyroid Allele-Specific Expression  
## GTEX.13QBU.0626.SM.5J2OG 3 Thyroid Allele-Specific Expression  
## GTEX.YFCO.0326.SM.4W1ZP 3 Thyroid RNA Seq (NGS)  
## sex Group ShortName sizeFactor  
## <factor> <factor> <factor> <numeric>  
## GTEX.S341.0226.SM.5S2VG male NIT S7SE-\_NIT 0.975311461378106  
## GTEX.ZAK1.0726.SM.5HL8Q female NIT ZC5H-\_NIT 0.898982680186865  
## GTEX.X15G.0526.SM.3NMB7 male NIT X4XY-\_NIT 0.944764423567596  
## GTEX.11DYG.0826.SM.5N9GH male NIT 11DZ1\_NIT 1.31916123634567  
## GTEX.Y5V6.0526.SM.4VBRV male NIT Y8E4-\_NIT 1.18547273258289  
## ... ... ... ... ...  
## GTEX.111FC.1026.SM.5GZX1 male ELI 111VG\_ELI 1.08969111335825  
## GTEX.TKQ1.0126.SM.33HB3 female ELI TMMY-\_ELI 1.68631884234377  
## GTEX.ZYW4.1126.SM.5SI99 female ELI ZYY3-\_ELI 0.938490380963618  
## GTEX.13QBU.0626.SM.5J2OG female ELI 13QJC\_ELI 0.933150561711403  
## GTEX.YFCO.0326.SM.4W1ZP male ELI YJ89-\_ELI 1.37509415513796

### Evaluación de clustering entre muestras:

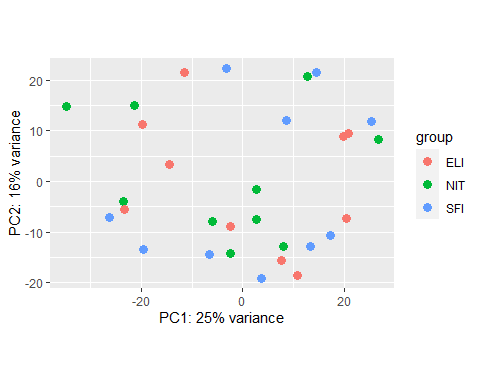
#### Cálculo de distancia entre muestras y representación mediante heatmap:

sampleDists <- dist(t(assay(vsd)))  
library("pheatmap")  
library("RColorBrewer")  
sampleDistMatrix <- as.matrix( sampleDists )  
rownames(sampleDistMatrix) <- paste(vsd$ShortName)  
colnames(sampleDistMatrix) <- NULL  
colors <- colorRampPalette( rev(brewer.pal(9, "Blues")) )(255)  
pheatmap(sampleDistMatrix,  
 clustering\_distance\_rows = sampleDists,  
 clustering\_distance\_cols = sampleDists,  
 col = colors)



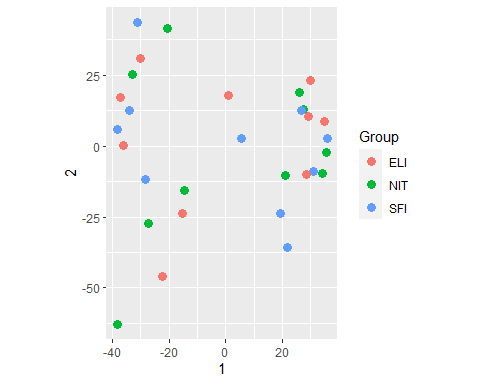
#### Análisis de componentes principales:

plotPCA(vsd, intgroup = c("Group"))



#### MDS plot:

mds <- as.data.frame(colData(vsd)) %>%  
 cbind(cmdscale(sampleDistMatrix))  
ggplot(mds, aes(x = `1`, y = `2`, color = Group)) +  
 geom\_point(size = 3) + coord\_fixed()



A priori, no se aprecia claramente ningún tipo de agrupamiento entre las muestras.

## Identificación de genes diferencialmente expresados

Se realizarán 3 comparaciones: SFI-NIT, ELI-NIT y ELI-SFI.

dds <- DESeq(dds, parallel =TRUE)

## estimating size factors

## estimating dispersions

## gene-wise dispersion estimates: 6 workers

## mean-dispersion relationship

## final dispersion estimates, fitting model and testing: 6 workers

## -- replacing outliers and refitting for 497 genes  
## -- DESeq argument 'minReplicatesForReplace' = 7   
## -- original counts are preserved in counts(dds)

## estimating dispersions

## fitting model and testing

### SFI-NIT

En primer lugar realizo el análisis y muestro los resultados del mismo

SFINITres <- results(dds, contrast=c("Group","SFI","NIT"))  
SFINITres

## log2 fold change (MLE): Group SFI vs NIT   
## Wald test p-value: Group SFI vs NIT   
## DataFrame with 43513 rows and 6 columns  
## baseMean log2FoldChange lfcSE  
## <numeric> <numeric> <numeric>  
## ENSG00000223972.4 4.53917358219955 0.321640332243348 0.690368689639189  
## ENSG00000227232.4 670.892244359886 0.117096934716329 0.187928076614192  
## ENSG00000243485.2 2.13920928094679 0.88755620472059 0.742624514175677  
## ENSG00000237613.2 1.88160561198927 -0.116680281105258 0.688835256002655  
## ENSG00000268020.2 1.05567101623418 0.976304510869576 0.944837192358173  
## ... ... ... ...  
## ENSG00000198695.2 76717.2806683366 -0.31085775938108 0.591644295502924  
## ENSG00000210194.1 41.4845000098372 -0.0714766875607364 0.818548217763293  
## ENSG00000198727.2 511745.795563112 0.230653383717484 0.281916766540973  
## ENSG00000210195.2 1.11108276000739 0.385610031964092 1.16676164055833  
## ENSG00000210196.2 3.53507751987634 -1.02930559926067 0.728002935952893  
## stat pvalue padj  
## <numeric> <numeric> <numeric>  
## ENSG00000223972.4 0.465896465280672 0.641289611391659 1  
## ENSG00000227232.4 0.623094413703406 0.533222480498503 1  
## ENSG00000243485.2 1.19516146824993 0.232023949836332 1  
## ENSG00000237613.2 -0.169387789153475 0.865491627566832 1  
## ENSG00000268020.2 1.03330448755183 0.301461427857032 1  
## ... ... ... ...  
## ENSG00000198695.2 -0.525413262231891 0.599295936037352 1  
## ENSG00000210194.1 -0.0873212915374107 0.930416130461572 1  
## ENSG00000198727.2 0.81816128408227 0.413265098968156 1  
## ENSG00000210195.2 0.330495980121156 0.741025229551141 1  
## ENSG00000210196.2 -1.4138756156435 0.157398426680564 1

summary(SFINITres)

##   
## out of 43507 with nonzero total read count  
## adjusted p-value < 0.1  
## LFC > 0 (up) : 0, 0%  
## LFC < 0 (down) : 1, 0.0023%  
## outliers [1] : 0, 0%  
## low counts [2] : 6, 0.014%  
## (mean count < 0)  
## [1] see 'cooksCutoff' argument of ?results  
## [2] see 'independentFiltering' argument of ?results

Considerando un p-valor ajustado de 0.1, muestro cuántos genes son significativamente diferentes en esta comparación:

table(SFINITres$padj < 0.1)

##   
## FALSE TRUE   
## 43506 1

A continuación, exraigo estos resultados y muestro los genes ordenados por mayor cambio en la expresión a la baja, es decir, que se expresan menos en NIT que en SFI:

SFINITresSig <- subset(SFINITres, padj < 0.1)  
SFINITdownreg <- subset(SFINITresSig,log2FoldChange<0)  
head(SFINITdownreg[ order(SFINITdownreg$log2FoldChange), ])

## log2 fold change (MLE): Group SFI vs NIT   
## Wald test p-value: Group SFI vs NIT   
## DataFrame with 1 row and 6 columns  
## baseMean log2FoldChange lfcSE  
## <numeric> <numeric> <numeric>  
## ENSG00000225972.1 2385.93178694658 -5.58136590652377 0.822759009483037  
## stat pvalue padj  
## <numeric> <numeric> <numeric>  
## ENSG00000225972.1 -6.78371897748127 1.17121048829752e-11 5.09558547143602e-07

Y por mayor a menor cambio en la expresión al alza, es decir, que se expresan mas en NIT que en SFI:

SFINITupreg <- subset(SFINITresSig,log2FoldChange>0)  
head(SFINITupreg[ order(SFINITupreg$log2FoldChange, decreasing = TRUE), ])

## log2 fold change (MLE): Group SFI vs NIT   
## Wald test p-value: Group SFI vs NIT   
## DataFrame with 0 rows and 6 columns

### ELI-NIT

En primer lugar realizo el análisis y muestro los resultados del mismo

ELINITres <- results(dds, contrast=c("Group","ELI","NIT"))  
ELINITres

## log2 fold change (MLE): Group ELI vs NIT   
## Wald test p-value: Group ELI vs NIT   
## DataFrame with 43513 rows and 6 columns  
## baseMean log2FoldChange lfcSE  
## <numeric> <numeric> <numeric>  
## ENSG00000223972.4 4.53917358219955 -0.512916872909814 0.70773340672622  
## ENSG00000227232.4 670.892244359886 0.302588865193538 0.187781284561297  
## ENSG00000243485.2 2.13920928094679 0.269809776460455 0.762961953075233  
## ENSG00000237613.2 1.88160561198927 -0.678619904675883 0.715390502654744  
## ENSG00000268020.2 1.05567101623418 -0.277234863810067 1.02177444753393  
## ... ... ... ...  
## ENSG00000198695.2 76717.2806683366 -0.193140646300286 0.59164374890405  
## ENSG00000210194.1 41.4845000098372 0.641515018367216 0.816917938053864  
## ENSG00000198727.2 511745.795563112 -0.0717923282623401 0.281916849924084  
## ENSG00000210195.2 1.11108276000739 -0.811121818990446 1.23076515995891  
## ENSG00000210196.2 3.53507751987634 -0.0918735124725379 0.692598175308102  
## stat pvalue padj  
## <numeric> <numeric> <numeric>  
## ENSG00000223972.4 -0.72473175356018 0.468616610871673 1  
## ENSG00000227232.4 1.61138989916093 0.107094767402009 1  
## ENSG00000243485.2 0.353634641115387 0.723612709931909 1  
## ENSG00000237613.2 -0.948600662376131 0.342823753997418 1  
## ENSG00000268020.2 -0.271326871090952 0.786139641204098 1  
## ... ... ... ...  
## ENSG00000198695.2 -0.326447539854949 0.744085774160543 1  
## ENSG00000210194.1 0.785286977411086 0.432285329357048 1  
## ENSG00000198727.2 -0.254657812336058 0.798987402996748 1  
## ENSG00000210195.2 -0.659038657722103 0.50987094488251 1  
## ENSG00000210196.2 -0.132650526305051 0.894469771259743 1

summary(ELINITres)

##   
## out of 43507 with nonzero total read count  
## adjusted p-value < 0.1  
## LFC > 0 (up) : 0, 0%  
## LFC < 0 (down) : 4, 0.0092%  
## outliers [1] : 0, 0%  
## low counts [2] : 6, 0.014%  
## (mean count < 0)  
## [1] see 'cooksCutoff' argument of ?results  
## [2] see 'independentFiltering' argument of ?results

Considerando un alpha de 0.05 y un p-valor ajustado de 0.1, muestro cuántos genes son significativamente diferentes en esta comparación:

table(ELINITres$padj < 0.1)

##   
## FALSE TRUE   
## 43503 4

A continuación, exraigo estos resultados y muestro los genes ordenados por mayor cambio en la expresión a la baja, es decir, que se expresan menos en ELI que en NIT:

ELINITresSig <- subset(ELINITres, padj < 0.1)  
ELINITdownreg <- subset(ELINITresSig,log2FoldChange<0)  
head(ELINITdownreg[ order(ELINITdownreg$log2FoldChange), ])

## log2 fold change (MLE): Group ELI vs NIT   
## Wald test p-value: Group ELI vs NIT   
## DataFrame with 4 rows and 6 columns  
## baseMean log2FoldChange lfcSE  
## <numeric> <numeric> <numeric>  
## ENSG00000225972.1 2385.93178694658 -5.91793364217918 0.822928077671824  
## ENSG00000099194.5 1465.01215402933 -3.48415209388505 0.782870861729024  
## ENSG00000016082.10 37.1448499331103 -2.90692051323182 0.640282345601159  
## ENSG00000125675.13 70.1140803007431 -2.62431047377788 0.58989127093801  
## stat pvalue padj  
## <numeric> <numeric> <numeric>  
## ENSG00000225972.1 -7.19131331491547 6.41710326487621e-13 2.79188911744969e-08  
## ENSG00000099194.5 -4.45048125330666 8.56780729380978e-06 0.0939206112036599  
## ENSG00000016082.10 -4.54006038617623 5.623811933063e-06 0.0939206112036599  
## ENSG00000125675.13 -4.44880370852897 8.63498850333601e-06 0.0939206112036599

Y por mayor a menor cambio en la expresión al alza, es decir, que se expresan mas en ELI que en NIT:

ELINITupreg <- subset(ELINITresSig,log2FoldChange>0)  
head(ELINITupreg[ order(ELINITupreg$log2FoldChange, decreasing = TRUE), ])

## log2 fold change (MLE): Group ELI vs NIT   
## Wald test p-value: Group ELI vs NIT   
## DataFrame with 0 rows and 6 columns

### ELI-SFI

En primer lugar realizo el análisis y muestro los resultados del mismo

ELISFIres <- results(dds, contrast=c("Group","ELI","SFI"))  
ELISFIres

## log2 fold change (MLE): Group ELI vs SFI   
## Wald test p-value: Group ELI vs SFI   
## DataFrame with 43513 rows and 6 columns  
## baseMean log2FoldChange lfcSE  
## <numeric> <numeric> <numeric>  
## ENSG00000223972.4 4.53917358219955 -0.834557205153162 0.699778252249183  
## ENSG00000227232.4 670.892244359886 0.185491930477209 0.187667342628895  
## ENSG00000243485.2 2.13920928094679 -0.617746428260135 0.717825952410277  
## ENSG00000237613.2 1.88160561198927 -0.561939623570626 0.717664397297725  
## ENSG00000268020.2 1.05567101623418 -1.25353937467964 0.953694238440694  
## ... ... ... ...  
## ENSG00000198695.2 76717.2806683366 0.117717113080794 0.591644157385173  
## ENSG00000210194.1 41.4845000098372 0.712991705927952 0.816940085428019  
## ENSG00000198727.2 511745.795563112 -0.302445711979825 0.281916702982957  
## ENSG00000210195.2 1.11108276000739 -1.19673185095454 1.20987533564232  
## ENSG00000210196.2 3.53507751987634 0.937432086788132 0.726389050863307  
## stat pvalue padj  
## <numeric> <numeric> <numeric>  
## ENSG00000223972.4 -1.19260237435328 0.233025135530346 0.774785570094762  
## ENSG00000227232.4 0.98840814751671 0.322952798879753 0.833646666275435  
## ENSG00000243485.2 -0.860579679775995 0.389469581510169 0.861997589699303  
## ENSG00000237613.2 -0.78301170531315 0.433620238348385 0.882181258118991  
## ENSG00000268020.2 -1.31440384575375 0.188710336978403 NA  
## ... ... ... ...  
## ENSG00000198695.2 0.198966070418164 0.842289285807328 0.979967784845527  
## ENSG00000210194.1 0.872758870137208 0.382794522262548 0.85895087348915  
## ENSG00000198727.2 -1.07281941360569 0.283352148632567 0.810402876391662  
## ENSG00000210195.2 -0.989136496711207 0.322596363168558 NA  
## ENSG00000210196.2 1.29053719308407 0.196864206866378 0.745421563280012

summary(ELISFIres)

##   
## out of 43507 with nonzero total read count  
## adjusted p-value < 0.1  
## LFC > 0 (up) : 35, 0.08%  
## LFC < 0 (down) : 34, 0.078%  
## outliers [1] : 0, 0%  
## low counts [2] : 14346, 33%  
## (mean count < 1)  
## [1] see 'cooksCutoff' argument of ?results  
## [2] see 'independentFiltering' argument of ?results

Considerando un alpha de 0.05 y un p-valor ajustado de 0.1, muestro cuántos genes son significativamente diferentes en esta comparación:

table(ELISFIres$padj < 0.1)

##   
## FALSE TRUE   
## 29098 69

A continuación, exraigo estos resultados y muestro los genes ordenados por mayor cambio en la expresión a la baja, es decir, que se expresan menos en ELI que en SFI (muestro solo los 5 primieros):

ELISFIresSig <- subset(ELISFIres, padj < 0.1)  
ELISFIdownreg <- subset(ELISFIresSig,log2FoldChange<0)  
head(ELISFIdownreg[ order(ELISFIdownreg$log2FoldChange), ], 5)

## log2 fold change (MLE): Group ELI vs SFI   
## Wald test p-value: Group ELI vs SFI   
## DataFrame with 5 rows and 6 columns  
## baseMean log2FoldChange lfcSE  
## <numeric> <numeric> <numeric>  
## ENSG00000110680.8 9969.14140220374 -6.11389361676835 1.57083356705019  
## ENSG00000259846.1 5.66201687165021 -5.43441311173801 1.46699905580045  
## ENSG00000256021.1 2.2469584142427 -3.39939571808126 0.827087315484252  
## ENSG00000223553.1 3.97793030212305 -2.93292319907057 0.749703481429823  
## ENSG00000016082.10 37.1448499331103 -2.88329855517083 0.640193741193381  
## stat pvalue padj  
## <numeric> <numeric> <numeric>  
## ENSG00000110680.8 -3.89213328834665 9.93666256985189e-05 0.0807149802224517  
## ENSG00000259846.1 -3.70444213324514 0.000211856462264458 0.0950648836133453  
## ENSG00000256021.1 -4.11008082754956 3.95520647766885e-05 0.0807149802224517  
## ENSG00000223553.1 -3.91211095015451 9.14928578157068e-05 0.0807149802224517  
## ENSG00000016082.10 -4.50379060219504 6.67519816894377e-06 0.0585708038035861

Y por mayor a menor cambio en la expresión al alza, es decir, que se expresan mas en ELI que en SFI, (muestro solo los 5 primieros):

ELISFIupreg <- subset(ELISFIresSig,log2FoldChange>0)  
head(ELISFIupreg[ order(ELISFIupreg$log2FoldChange, decreasing = TRUE), ],5)

## log2 fold change (MLE): Group ELI vs SFI   
## Wald test p-value: Group ELI vs SFI   
## DataFrame with 5 rows and 6 columns  
## baseMean log2FoldChange lfcSE  
## <numeric> <numeric> <numeric>  
## ENSG00000092758.11 6110.67006019468 2.79825944117508 0.695919165158303  
## ENSG00000225383.2 51.7243663525045 2.5143660889107 0.675445544035939  
## ENSG00000135069.9 1012.22959924315 2.1204237690974 0.529426250652791  
## ENSG00000150051.9 72.9236542320438 1.67445367689069 0.429322484459516  
## ENSG00000226944.1 29.8204088557369 1.09619283752744 0.297968533231977  
## stat pvalue padj  
## <numeric> <numeric> <numeric>  
## ENSG00000092758.11 4.02095470461508 5.79627505510231e-05 0.0807149802224517  
## ENSG00000225383.2 3.72252968594152 0.000197236710561924 0.0950648836133453  
## ENSG00000135069.9 4.00513530729329 6.19819919309922e-05 0.0807149802224517  
## ENSG00000150051.9 3.90022357901589 9.61038963632936e-05 0.0807149802224517  
## ENSG00000226944.1 3.67888792026915 0.000234253178634445 0.0999457549054777

## Anotación de los resultados:

Dado que el nombre ENSEMBL de los genes es poco informativo, es conveniente anotar los resultados para una mejor interpretación.

### SFI-NIT

# En primer lugar ajusto la nomenclatura a la adecuada para llevar a cabo la anotación  
rownames(SFINITres) <- gsub("\\..\*", "", rownames(SFINITres), fixed = FALSE)  
# Tras esto ya puedo realizar la anotación correctamente  
library("AnnotationDbi")  
SFINITres$symbol <- mapIds(org.Hs.eg.db,  
 keys = rownames(SFINITres),  
 column = "SYMBOL",  
 keytype = "ENSEMBL",  
 multiVals="first")

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

SFINITres$entrez <- mapIds(org.Hs.eg.db,  
 keys = rownames(SFINITres),  
 column = "ENTREZID",  
 keytype = "ENSEMBL",  
 multiVals="first")

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

SFINITresOrdered <- SFINITres[order(SFINITres$pvalue),]  
head(SFINITresOrdered)

## log2 fold change (MLE): Group SFI vs NIT   
## Wald test p-value: Group SFI vs NIT   
## DataFrame with 6 rows and 8 columns  
## baseMean log2FoldChange lfcSE  
## <numeric> <numeric> <numeric>  
## ENSG00000225972 2385.93178694658 -5.58136590652377 0.822759009483037  
## ENSG00000099866 101.739204947545 -1.45779463220455 0.356960783721761  
## ENSG00000143546 519.384545188698 1.27948272624513 0.328656390015101  
## ENSG00000205592 3.4283761196695 3.88641375706823 1.11382580328716  
## ENSG00000231205 248.152120026799 -0.799538930671605 0.231295559203946  
## ENSG00000233942 8.21808507693208 -1.8703762146968 0.542326447664897  
## stat pvalue padj  
## <numeric> <numeric> <numeric>  
## ENSG00000225972 -6.78371897748127 1.17121048829752e-11 5.09558547143602e-07  
## ENSG00000099866 -4.08390696872978 4.42847659586296e-05 0.963348656281048  
## ENSG00000143546 3.89307119872623 9.89830861532751e-05 1  
## ENSG00000205592 3.48924737207425 0.000484382668293054 1  
## ENSG00000231205 -3.4567846154219 0.000546661588564391 1  
## ENSG00000233942 -3.44880140503955 0.000563080663701698 1  
## symbol entrez  
## <character> <character>  
## ENSG00000225972 NA NA  
## ENSG00000099866 MADCAM1 8174  
## ENSG00000143546 S100A8 6279  
## ENSG00000205592 MUC19 283463  
## ENSG00000231205 NA NA  
## ENSG00000233942 NA NA

### ELI-NIT

# En primer lugar ajusto la nomenclatura a la adecuada para llevar a cabo la anotación  
rownames(ELINITres) <- gsub("\\..\*", "", rownames(ELINITres), fixed = FALSE)  
# Tras esto ya puedo realizar la anotación correctamente  
library("AnnotationDbi")  
ELINITres$symbol <- mapIds(org.Hs.eg.db,  
 keys = rownames(ELINITres),  
 column = "SYMBOL",  
 keytype = "ENSEMBL",  
 multiVals="first")

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

ELINITres$entrez <- mapIds(org.Hs.eg.db,  
 keys = rownames(ELINITres),  
 column = "ENTREZID",  
 keytype = "ENSEMBL",  
 multiVals="first")

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

ELINITresOrdered <- ELINITres[order(ELINITres$pvalue),]  
head(ELINITresOrdered)

## 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  
## <numeric> <numeric> <numeric>  
## ENSG00000225972 2385.93178694658 -5.91793364217918 0.822928077671824  
## ENSG00000016082 37.1448499331103 -2.90692051323182 0.640282345601159  
## ENSG00000099194 1465.01215402933 -3.48415209388505 0.782870861729024  
## ENSG00000125675 70.1140803007431 -2.62431047377788 0.58989127093801  
## ENSG00000162105 2725.69481826633 0.585152084983046 0.139376023225019  
## ENSG00000089199 408.423759865464 -3.72095095262579 0.8957264841416  
## stat pvalue padj  
## <numeric> <numeric> <numeric>  
## ENSG00000225972 -7.19131331491547 6.41710326487621e-13 2.79188911744969e-08  
## ENSG00000016082 -4.54006038617623 5.623811933063e-06 0.0939206112036599  
## ENSG00000099194 -4.45048125330666 8.56780729380978e-06 0.0939206112036599  
## ENSG00000125675 -4.44880370852897 8.63498850333601e-06 0.0939206112036599  
## ENSG00000162105 4.19836978730792 2.68843368408977e-05 0.233931368587388  
## ENSG00000089199 -4.15411514396795 3.26548718661252e-05 0.236139498811211  
## symbol entrez  
## <character> <character>  
## ENSG00000225972 NA NA  
## ENSG00000016082 ISL1 3670  
## ENSG00000099194 SCD 6319  
## ENSG00000125675 GRIA3 2892  
## ENSG00000162105 SHANK2 22941  
## ENSG00000089199 CHGB 1114

### ELI-SFI

# En primer lugar ajusto la nomenclatura a la adecuada para llevar a cabo la anotación  
rownames(ELISFIres) <- gsub("\\..\*", "", rownames(ELISFIres), fixed = FALSE)  
# Tras esto ya puedo realizar la anotación correctamente  
library("AnnotationDbi")  
ELISFIres$symbol <- mapIds(org.Hs.eg.db,  
 keys = rownames(ELISFIres),  
 column = "SYMBOL",  
 keytype = "ENSEMBL",  
 multiVals="first")

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

ELISFIres$entrez <- mapIds(org.Hs.eg.db,  
 keys = rownames(ELISFIres),  
 column = "ENTREZID",  
 keytype = "ENSEMBL",  
 multiVals="first")

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

ELISFIresOrdered <- ELISFIres[order(ELISFIres$pvalue),]  
head(ELISFIresOrdered)

## 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  
## <numeric> <numeric> <numeric>  
## ENSG00000065882 7519.86791630768 0.355919705836386 0.0715804667311189  
## ENSG00000139168 1592.24479322926 0.552703167797859 0.120617389305866  
## ENSG00000016082 37.1448499331103 -2.88329855517083 0.640193741193381  
## ENSG00000231205 248.152120026799 1.03074857243472 0.230886088164035  
## ENSG00000254165 77.2711553569672 -1.00925708830735 0.23224524864338  
## ENSG00000232027 6.80131678992827 -1.69078693318295 0.394873323222044  
## stat pvalue padj  
## <numeric> <numeric> <numeric>  
## ENSG00000065882 4.97230211104022 6.61624932620464e-07 0.0192976144097411  
## ENSG00000139168 4.58228428735341 4.59924047806455e-06 0.0585708038035861  
## ENSG00000016082 -4.50379060219504 6.67519816894377e-06 0.0585708038035861  
## ENSG00000231205 4.4643164975034 8.03247557905662e-06 0.0585708038035861  
## ENSG00000254165 -4.34565225425599 1.38862412545849e-05 0.0807149802224517  
## ENSG00000232027 -4.28184643973073 1.85348877105353e-05 0.0807149802224517  
## symbol entrez  
## <character> <character>  
## ENSG00000065882 TBC1D1 23216  
## ENSG00000139168 ZCRB1 85437  
## ENSG00000016082 ISL1 3670  
## ENSG00000231205 NA NA  
## ENSG00000254165 NA NA  
## ENSG00000232027 NA NA

## Exportación de los datos:

SFINITresOrderedDF <- as.data.frame(SFINITresOrdered)  
write.csv(SFINITresOrderedDF, file = "resultsSFI\_NIT.csv")  
ELINITresOrderedDF <- as.data.frame(ELINITresOrdered)  
write.csv(ELINITresOrderedDF, file = "resultsELI\_NIT.csv")  
ELISFIresOrderedDF <- as.data.frame(ELISFIresOrdered)  
write.csv(ELISFIresOrderedDF, file = "resultsELI\_SFI.csv")