Análisis de Datos Ómicos. Juan Carlos Prieto Prieto. PEC 2

Prieto Prieto, Juan Carlos

1/6/2020

Repositorio de GitHub: <https://github.com/juancp347/Prieto-Juan-Carlos-ADO-PEC-2>

### 1) Definición de los datos

Comenzaremos importando los dos archivos csv en nuestra sesión:

targets <- read.csv("C:/Users/usuario/Desktop/Master/3er Semestre/Análisis datos ómicos/PEC2/Prieto-Juan-Carlos-ADO-PEC-2/targets.csv")  
dim(targets)

## [1] 292 9

counts <- read.csv("C:/Users/usuario/Desktop/Master/3er Semestre/Análisis datos ómicos/PEC2/Prieto-Juan-Carlos-ADO-PEC-2/counts.csv", header=T, sep=";", row.names = 1)  
dim(counts)

## [1] 56202 292

A continuación creamos una muestra aleatoria de targets mediante la función strata(). Fijamos una semilla para que los datos sean reproducibles.

set.seed(123456)  
library(sampling)  
muestra <- strata(targets, stratanames = c("Group"), size = c(10,10,10), method = "srswor")

Con la muestra aleatoria que hemos obtenido con la función strata(), creamos nuestro nuevo dataframe con todas las variables de targets:

muestra2 <- data.frame(targets[c(31, 53, 62, 107, 111, 120, 155, 225, 237, 280, 29, 40, 100, 149, 167, 186, 211, 251, 253, 290, 22, 30, 37, 83, 171, 188, 199, 203, 261, 288),])

A continuación seleccionamos el subset del csv counts:

counts2 <- counts[,c(31, 53, 62, 107, 111, 120, 155, 225, 237, 280, 29, 40, 100, 149, 167, 186, 211, 251, 253, 290, 22, 30, 37, 83, 171, 188, 199, 203, 261, 288)]

counts3 <- t(counts2)

df <- cbind(muestra2, counts3)  
df2 <- t(counts3)

### 2) Control de calidad de los datos crudos

Con la función DGEList eliminamos los genes con contaje 0:

library(edgeR)  
group <- c(rep("NIT",10), rep("ELI",10), rep("SFI", 10))  
data1 <- DGEList(df2,group=group,remove.zeros=TRUE)

## Removing 9370 rows with all zero counts

data1

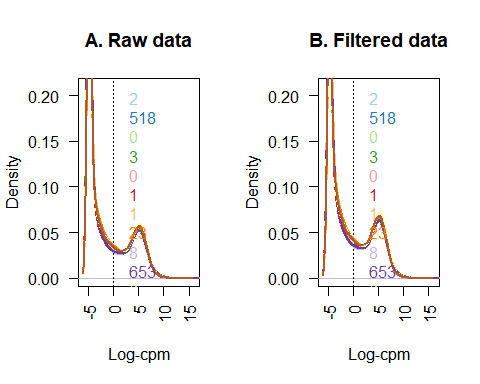
## An object of class "DGEList"  
## $counts  
## GTEX.11OF3.0626.SM.5BC4Y GTEX.12WSH.0226.SM.5GCOG  
## ENSG00000223972.4 2 3  
## ENSG00000227232.4 518 641  
## ENSG00000243485.2 0 0  
## ENSG00000237613.2 3 0  
## ENSG00000268020.2 0 0  
## GTEX.13112.0326.SM.5P9IW GTEX.13O61.0226.SM.5KM52  
## ENSG00000223972.4 3 2  
## ENSG00000227232.4 600 624  
## ENSG00000243485.2 1 1  
## ENSG00000237613.2 0 1  
## ENSG00000268020.2 4 2  
## GTEX.13OW5.0626.SM.5J2N2 GTEX.13RTK.0326.SM.5RQHS  
## ENSG00000223972.4 1 0  
## ENSG00000227232.4 1042 627  
## ENSG00000243485.2 1 0  
## ENSG00000237613.2 4 0  
## ENSG00000268020.2 0 1  
## GTEX.OHPK.2626.SM.2HMK9 GTEX.X15G.0526.SM.3NMB7  
## ENSG00000223972.4 17 26  
## ENSG00000227232.4 484 810  
## ENSG00000243485.2 10 6  
## ENSG00000237613.2 7 7  
## ENSG00000268020.2 6 4  
## GTEX.XYKS.0826.SM.4BRVF GTEX.ZVZP.1026.SM.5GICI  
## ENSG00000223972.4 4 5  
## ENSG00000227232.4 432 529  
## ENSG00000243485.2 0 2  
## ENSG00000237613.2 1 1  
## ENSG00000268020.2 1 1  
## GTEX.11NV4.0626.SM.5N9BR GTEX.11XUK.0226.SM.5EQLW  
## ENSG00000223972.4 3 0  
## ENSG00000227232.4 1301 419  
## ENSG00000243485.2 1 0  
## ENSG00000237613.2 0 1  
## ENSG00000268020.2 0 0  
## GTEX.13NZ9.1126.SM.5MR37 GTEX.14BMU.0226.SM.5S2QA  
## ENSG00000223972.4 0 2  
## ENSG00000227232.4 1002 423  
## ENSG00000243485.2 1 0  
## ENSG00000237613.2 0 0  
## ENSG00000268020.2 0 2  
## GTEX.PLZ4.1226.SM.2I5FE GTEX.R55G.0726.SM.2TC6J  
## ENSG00000223972.4 5 3  
## ENSG00000227232.4 489 134  
## ENSG00000243485.2 1 1  
## ENSG00000237613.2 3 2  
## ENSG00000268020.2 2 1  
## GTEX.TMMY.0826.SM.33HB9 GTEX.YFC4.2626.SM.5P9FQ  
## ENSG00000223972.4 3 1  
## ENSG00000227232.4 979 1472  
## ENSG00000243485.2 3 1  
## ENSG00000237613.2 2 0  
## ENSG00000268020.2 5 0  
## GTEX.YJ89.0726.SM.5P9F7 GTEX.ZYY3.1926.SM.5GZXS  
## ENSG00000223972.4 4 6  
## ENSG00000227232.4 1325 1003  
## ENSG00000243485.2 1 1  
## ENSG00000237613.2 0 2  
## ENSG00000268020.2 2 0  
## GTEX.11EQ9.0626.SM.5A5K1 GTEX.11O72.2326.SM.5BC7H  
## ENSG00000223972.4 6 0  
## ENSG00000227232.4 640 633  
## ENSG00000243485.2 4 2  
## ENSG00000237613.2 3 1  
## ENSG00000268020.2 1 0  
## GTEX.11TUW.0226.SM.5LU8X GTEX.13FH7.0126.SM.5KLZ1  
## ENSG00000223972.4 4 5  
## ENSG00000227232.4 627 576  
## ENSG00000243485.2 0 4  
## ENSG00000237613.2 1 3  
## ENSG00000268020.2 0 0  
## GTEX.Q2AH.0726.SM.2I3EA GTEX.RM2N.0526.SM.2TF4N  
## ENSG00000223972.4 1 3  
## ENSG00000227232.4 874 406  
## ENSG00000243485.2 8 4  
## ENSG00000237613.2 2 1  
## ENSG00000268020.2 0 0  
## GTEX.S341.0226.SM.5S2VG GTEX.SIU8.0626.SM.2XCDN  
## ENSG00000223972.4 2 3  
## ENSG00000227232.4 457 899  
## ENSG00000243485.2 1 2  
## ENSG00000237613.2 2 2  
## ENSG00000268020.2 0 0  
## GTEX.ZE7O.1126.SM.57WC8 GTEX.ZYVF.1126.SM.5E458  
## ENSG00000223972.4 1 2  
## ENSG00000227232.4 713 838  
## ENSG00000243485.2 2 1  
## ENSG00000237613.2 0 4  
## ENSG00000268020.2 0 1  
## 46827 more rows ...  
##   
## $samples  
## group lib.size norm.factors  
## GTEX.11OF3.0626.SM.5BC4Y NIT 50681092 1  
## GTEX.12WSH.0226.SM.5GCOG NIT 52544311 1  
## GTEX.13112.0326.SM.5P9IW NIT 65440872 1  
## GTEX.13O61.0226.SM.5KM52 NIT 58497774 1  
## GTEX.13OW5.0626.SM.5J2N2 NIT 70246157 1  
## 25 more rows ...

Y realizamos una comparación gráfica de ambos grupos de datos, comparando mediante los valores de Log-cpm:

library(RColorBrewer)  
lcpm <- cpm(df2, log=TRUE)  
nsamples <- ncol(df2)  
col <- brewer.pal(nsamples, "Paired")

## Warning in brewer.pal(nsamples, "Paired"): n too large, allowed maximum for palette Paired is 12  
## Returning the palette you asked for with that many colors

par(mfrow=c(1,2))  
plot(density(lcpm[,1]), col=col[1], lwd=2, ylim=c(0,0.21), las=2, main="", xlab="")  
title(main="A. Raw data", xlab="Log-cpm")  
abline(v=0, lty=3)  
for (i in 2:nsamples){  
 den <- density(lcpm[,i])  
 lines(den$x, den$y, col=col[i], lwd=2)  
}  
legend(x="topright", legend=data1$counts, text.col=col, bty="n")  
  
lcpm <- cpm(data1, log=TRUE)  
plot(density(lcpm[,1]), col=col[1], lwd=2, ylim=c(0,0.21), las=2,   
 main="", xlab="")  
title(main="B. Filtered data", xlab="Log-cpm")  
abline(v=0, lty=3)  
for (i in 2:nsamples){  
 den <- density(lcpm[,i])  
 lines(den$x, den$y, col=col[i], lwd=2)  
}  
legend(x="topright", legend=data1$counts, text.col=col, bty="n")



### 3) Normalización:

El paso de la normalización debemos realizarlo para que los datos de los genes sean comparables, pues puede que algunas lecturas sean mayores que otras (en diferentes genes), y por tanto no serían comparables.

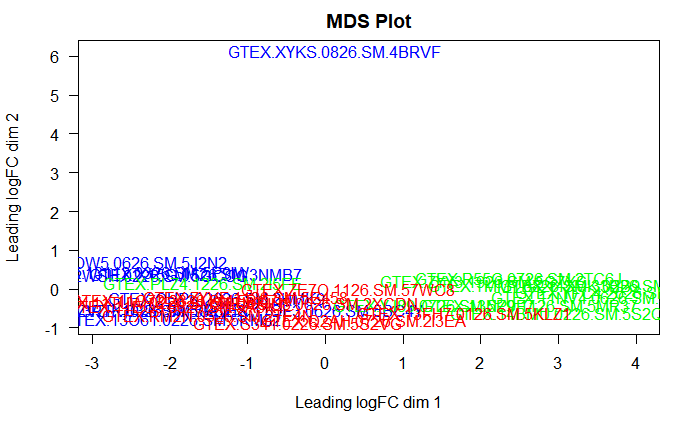
data2 <- calcNormFactors(data1)  
data2

## An object of class "DGEList"  
## $counts  
## GTEX.11OF3.0626.SM.5BC4Y GTEX.12WSH.0226.SM.5GCOG  
## ENSG00000223972.4 2 3  
## ENSG00000227232.4 518 641  
## ENSG00000243485.2 0 0  
## ENSG00000237613.2 3 0  
## ENSG00000268020.2 0 0  
## GTEX.13112.0326.SM.5P9IW GTEX.13O61.0226.SM.5KM52  
## ENSG00000223972.4 3 2  
## ENSG00000227232.4 600 624  
## ENSG00000243485.2 1 1  
## ENSG00000237613.2 0 1  
## ENSG00000268020.2 4 2  
## GTEX.13OW5.0626.SM.5J2N2 GTEX.13RTK.0326.SM.5RQHS  
## ENSG00000223972.4 1 0  
## ENSG00000227232.4 1042 627  
## ENSG00000243485.2 1 0  
## ENSG00000237613.2 4 0  
## ENSG00000268020.2 0 1  
## GTEX.OHPK.2626.SM.2HMK9 GTEX.X15G.0526.SM.3NMB7  
## ENSG00000223972.4 17 26  
## ENSG00000227232.4 484 810  
## ENSG00000243485.2 10 6  
## ENSG00000237613.2 7 7  
## ENSG00000268020.2 6 4  
## GTEX.XYKS.0826.SM.4BRVF GTEX.ZVZP.1026.SM.5GICI  
## ENSG00000223972.4 4 5  
## ENSG00000227232.4 432 529  
## ENSG00000243485.2 0 2  
## ENSG00000237613.2 1 1  
## ENSG00000268020.2 1 1  
## GTEX.11NV4.0626.SM.5N9BR GTEX.11XUK.0226.SM.5EQLW  
## ENSG00000223972.4 3 0  
## ENSG00000227232.4 1301 419  
## ENSG00000243485.2 1 0  
## ENSG00000237613.2 0 1  
## ENSG00000268020.2 0 0  
## GTEX.13NZ9.1126.SM.5MR37 GTEX.14BMU.0226.SM.5S2QA  
## ENSG00000223972.4 0 2  
## ENSG00000227232.4 1002 423  
## ENSG00000243485.2 1 0  
## ENSG00000237613.2 0 0  
## ENSG00000268020.2 0 2  
## GTEX.PLZ4.1226.SM.2I5FE GTEX.R55G.0726.SM.2TC6J  
## ENSG00000223972.4 5 3  
## ENSG00000227232.4 489 134  
## ENSG00000243485.2 1 1  
## ENSG00000237613.2 3 2  
## ENSG00000268020.2 2 1  
## GTEX.TMMY.0826.SM.33HB9 GTEX.YFC4.2626.SM.5P9FQ  
## ENSG00000223972.4 3 1  
## ENSG00000227232.4 979 1472  
## ENSG00000243485.2 3 1  
## ENSG00000237613.2 2 0  
## ENSG00000268020.2 5 0  
## GTEX.YJ89.0726.SM.5P9F7 GTEX.ZYY3.1926.SM.5GZXS  
## ENSG00000223972.4 4 6  
## ENSG00000227232.4 1325 1003  
## ENSG00000243485.2 1 1  
## ENSG00000237613.2 0 2  
## ENSG00000268020.2 2 0  
## GTEX.11EQ9.0626.SM.5A5K1 GTEX.11O72.2326.SM.5BC7H  
## ENSG00000223972.4 6 0  
## ENSG00000227232.4 640 633  
## ENSG00000243485.2 4 2  
## ENSG00000237613.2 3 1  
## ENSG00000268020.2 1 0  
## GTEX.11TUW.0226.SM.5LU8X GTEX.13FH7.0126.SM.5KLZ1  
## ENSG00000223972.4 4 5  
## ENSG00000227232.4 627 576  
## ENSG00000243485.2 0 4  
## ENSG00000237613.2 1 3  
## ENSG00000268020.2 0 0  
## GTEX.Q2AH.0726.SM.2I3EA GTEX.RM2N.0526.SM.2TF4N  
## ENSG00000223972.4 1 3  
## ENSG00000227232.4 874 406  
## ENSG00000243485.2 8 4  
## ENSG00000237613.2 2 1  
## ENSG00000268020.2 0 0  
## GTEX.S341.0226.SM.5S2VG GTEX.SIU8.0626.SM.2XCDN  
## ENSG00000223972.4 2 3  
## ENSG00000227232.4 457 899  
## ENSG00000243485.2 1 2  
## ENSG00000237613.2 2 2  
## ENSG00000268020.2 0 0  
## GTEX.ZE7O.1126.SM.57WC8 GTEX.ZYVF.1126.SM.5E458  
## ENSG00000223972.4 1 2  
## ENSG00000227232.4 713 838  
## ENSG00000243485.2 2 1  
## ENSG00000237613.2 0 4  
## ENSG00000268020.2 0 1  
## 46827 more rows ...  
##   
## $samples  
## group lib.size norm.factors  
## GTEX.11OF3.0626.SM.5BC4Y NIT 50681092 1.0382047  
## GTEX.12WSH.0226.SM.5GCOG NIT 52544311 0.9466910  
## GTEX.13112.0326.SM.5P9IW NIT 65440872 1.0996362  
## GTEX.13O61.0226.SM.5KM52 NIT 58497774 0.8516618  
## GTEX.13OW5.0626.SM.5J2N2 NIT 70246157 0.9781575  
## 25 more rows ...

La columna “norm.factors” ha cambiado sus valores, por lo que observamos que la normalización ha sido efectiva.

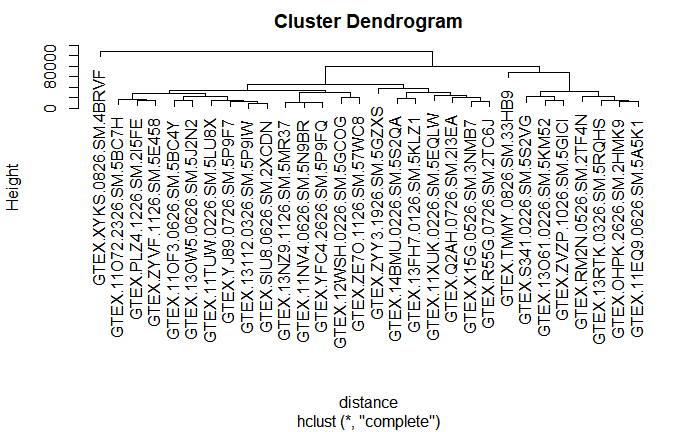
### Control de calidad de la normalización:

library(limma)  
nit.color <-"blue"  
eli.color <- "green"  
sfi.color <- "red"  
main <- "MDS Plot"  
colors <- c(rep(nit.color,10), rep(eli.color,10), rep(sfi.color,10))  
plotMDS(data1, main=main, labels=colnames(data1$counts), col=colors,las=1)



Apreciamos que existe una cierta diferenciación entre los tres grupos de comparación, y resulta llamativa la diferenciación de una de las muestras del grupo NIT respecto de las otras 29 muestras.

normalized.counts=cpm(data2)  
transposed=t(normalized.counts)  
distance=dist(transposed)  
clusters=hclust(distance)  
plot(clusters)



Esta gráfica nos muestra un resultado similar que ya veíamos en la gráfica MDS, por tanto los resultados de ambas son congruentes, mostrando similitudes entre los mismas muestras.

### 4) Identificación de genes diferencialmente expresados:

data1 <- estimateCommonDisp(data1)  
data1 <- estimateTagwiseDisp(data1)

Realizamos una a una las comparaciones entre los diferentes grupos: NIT-ELI, NIT-SFI y ELI-SFI

dex <- exactTest(data1, pair=c("NIT","ELI"), dispersion="tagwise")  
  
fdrvalues <- p.adjust(dex$table$PValue, method="BH")  
dex$table$fdr <- fdrvalues  
  
summary(decideTestsDGE(dex,p=0.05))

## ELI-NIT  
## Down 740  
## NotSig 42947  
## Up 3145

dex2 <- exactTest(data1, pair=c("NIT","SFI"), dispersion="tagwise")  
  
fdrvalues <- p.adjust(dex2$table$PValue, method="BH")  
dex2$table$fdr <- fdrvalues  
  
summary(decideTestsDGE(dex2,p=0.05))

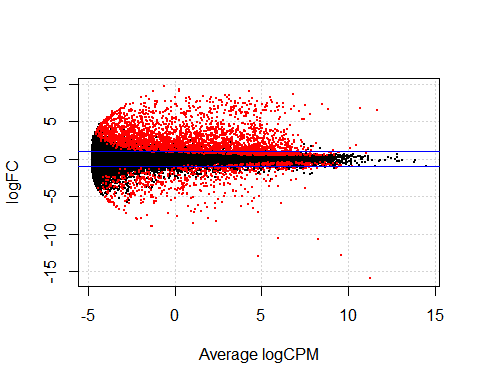
## SFI-NIT  
## Down 275  
## NotSig 46441  
## Up 116

dex3 <- exactTest(data1, pair=c("ELI","SFI"), dispersion="tagwise")  
  
fdrvalues <- p.adjust(dex3$table$PValue, method="BH")  
dex3$table$fdr <- fdrvalues  
  
summary(decideTestsDGE(dex3,p=0.05))

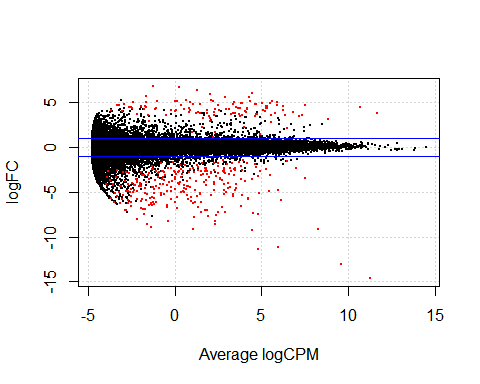
## SFI-ELI  
## Down 2619  
## NotSig 43804  
## Up 409

Y representamos gráficamente los resultados:

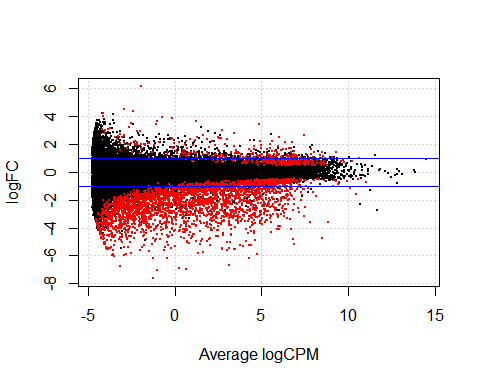
cutoff <- 0.05  
de <- decideTestsDGE(dex, p = cutoff, adjust = "BH")  
detags <- rownames(dex)[as.logical(de)]  
plotSmear(dex, de.tags = detags)  
abline(h = c(-1, 1), col = "blue")



de2 <- decideTestsDGE(dex2, p = cutoff, adjust = "BH")  
detags2 <- rownames(dex2)[as.logical(de2)]  
plotSmear(dex2, de.tags = detags2)  
abline(h = c(-1, 1), col = "blue")



de3 <- decideTestsDGE(dex3, p = cutoff, adjust = "BH")  
detags3 <- rownames(dex3)[as.logical(de3)]  
plotSmear(dex3, de.tags = detags3)  
abline(h = c(-1, 1), col = "blue")



Obtenemos los valores (en rojo) de sobre/infraexpresión. De manera aproximada observamos resultados similares al valor del número de genes up/down-regulated.

### 5) Anotación de resultados