Analise RNASeq a celulas humanas TNBC alinhadas ao cromossoma 13

Grupo 2

14 de Junho 2019

```r
#load de packages necessarios
dependencias <- c(
  'edgeR',
  'limma',
  'Glimma',
  'gplots',
  'org.Mm.eg.db',
  'RColorBrewer',
  'DESeq2',
  'pheatmap',
  'RColorBrewer'
)

invisible(suppressMessages(
  lapply(
    dependencias,
    library,
    character.only = T,
    warn.conflicts = FALSE,
    quietly = TRUE
  )
))
```

```
## Warning: package 'gplots' was built under R version 3.5.3
```

```
## Warning: package 'matrixStats' was built under R version 3.5.3
```

```
## Warning: package 'pheatmap' was built under R version 3.5.3
```

```r
#load de tabela de contagens
sr13 <- read.table("ch13finalreadcount.tab", h = T, row.names = 1)
tail(sr13) # fragmentos do ficheiro que nao sao necessarios para a analise
```

```
##                             SR05    SR06    SR07    SR08
## ENSG00000286272                0       0       0       0
## __no_feature                5339    6723    6447    5270
## __ambiguous                  586     752     673     507
## __too_low_aQual             6295    7845    7440    6634
## __not_aligned            3373953 4125118 4214933 3694022
## __alignment_not_unique         0       0       0       0
```

```r
#remover fragmentos
sr13 <- sr13[1:(nrow(sr13) - 5), ]
head(sr13)
```

```
##                 SR05 SR06 SR07 SR08
## ENSG00000000003    0    0    0    0
## ENSG00000000005    0    0    0    0
## ENSG00000000419    0    0    0    0
## ENSG00000000457    0    0    0    0
## ENSG00000000460    0    0    0    0
## ENSG00000000938    0    0    0    0
```

```r
#filtrar genes que nao sao expressos
sr13 <- sr13[rowSums(sr13) > 1,]
head(sr13)
```

```
##                   SR05 SR06 SR07 SR08
## ENSG00000005810   70   83   66  110
## ENSG00000005812   23   16   16   35
## ENSG00000023516   13   11   12   10
## ENSG00000027001    1    1    3    1
## ENSG00000032742    2    4    2    4
## ENSG00000065150  232  337  304  145
```

```r
dim(sr13) #dimensoes apos aplicar filtro
```

```
## [1] 393   4
```

```r
#definir fatores das amostras para analise com DESeq2
condition <- factor(c("bulk", "bulk", "bulk", "spheroid"))
cd = data.frame(c("bulk", "bulk", "bulk", "spheroid"))
colnames(cd)[1] = "condition"
rownames(cd) = colnames(sr13)
```

```r
# Analise Diferencial
dds <- DESeqDataSetFromMatrix(countData = sr13,
                              colData = cd,
                              design = ~ condition)
dds <- DESeq(dds)
```

```
## estimating size factors

## estimating dispersions

## gene-wise dispersion estimates

## mean-dispersion relationship

## final dispersion estimates

## fitting model and testing
```

```r
res <- results(dds)
res
```

```
## log2 fold change (MLE): condition spheroid vs bulk
## Wald test p-value: condition spheroid vs bulk
## DataFrame with 393 rows and 6 columns
##                         baseMean     log2FoldChange             lfcSE
##                        <numeric>          <numeric>         <numeric>
## ENSG00000005810 83.3369320468701  0.963171729333744 0.381698743139506
## ENSG00000005812 23.4180500576859   1.28111649494208 0.718492324093415
## ENSG00000023516 11.3592808474381 0.0996065631819345  1.01883551662264
## ENSG00000027001 1.40633695862894 -0.320715605154776   2.8644118291177
## ENSG00000032742 3.01339409971724  0.971031848007647  1.89214470647887
## ...                          ...                ...               ...
## ENSG00000279231 4.05278699654957   1.23906806225683  1.72518136603546
## ENSG00000279730  190.258999386735  0.225826192817722 0.255299315534446
## ENSG00000280060  1.12785621272685  -2.78220211817549  3.51325537776182
## ENSG00000281106 0.803030659536842   2.85551999990813  3.85235454246864
## ENSG00000284196 0.505136052760395   1.85551748353773  4.76635002388493
##                               stat             pvalue               padj
##                          <numeric>          <numeric>          <numeric>
## ENSG00000005810   2.52338197766011 0.0116232069503501 0.0431719115298716
## ENSG00000005812   1.78306218728026 0.0745761808402859                 NA
## ENSG00000023516  0.097765106886068  0.922118815281034                 NA
## ENSG00000027001 -0.111965605606916  0.910850677796894                 NA
## ENSG00000032742  0.513191113070132  0.607817641831496                 NA
## ...                            ...                ...                ...
## ENSG00000279231  0.718224811982677  0.472618681512135                 NA
## ENSG00000279730  0.884554634801803  0.376396883226328  0.559218226507687
## ENSG00000280060 -0.791915707519087  0.428409827503433                 NA
## ENSG00000281106  0.741240186599825  0.458547821075764                 NA
## ENSG00000284196  0.389295262462774  0.697057740461111                 NA
```

```
mcols(res, use.names = TRUE) #metadados para legenda
```

```
## DataFrame with 6 rows and 2 columns
##                        type
##                 <character>
## baseMean       intermediate
## log2FoldChange      results
## lfcSE               results
## stat                results
## pvalue              results
## padj                results
##                                                        description
##                                                        <character>
## baseMean                mean of normalized counts for all samples
## log2FoldChange log2 fold change (MLE): condition spheroid vs bulk
## lfcSE                  standard error: condition spheroid vs bulk
## stat                   Wald statistic: condition spheroid vs bulk
## pvalue              Wald test p-value: condition spheroid vs bulk
## padj                                         BH adjusted p-values
```

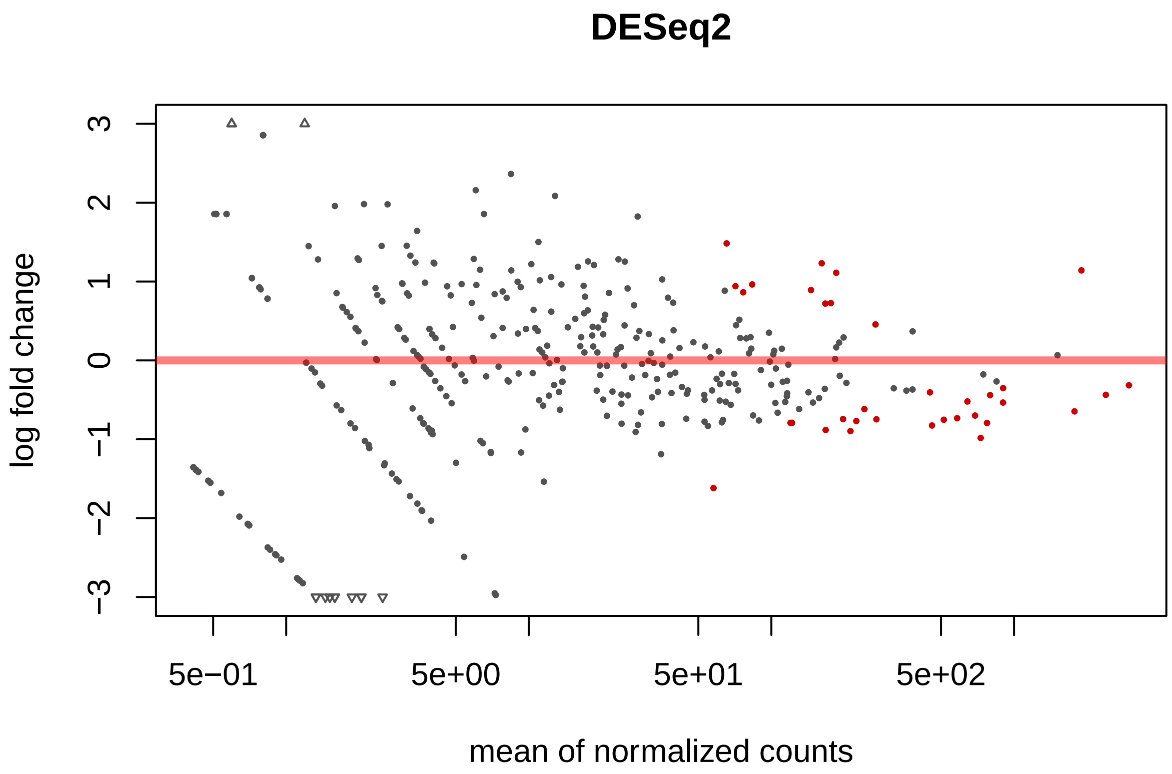
```
resOrdered <- res[order(res$padj),]
summary(res)
```

```
##
## out of 393 with nonzero total read count
## adjusted p-value < 0.1
## LFC > 0 (up)       : 11, 2.8%
## LFC < 0 (down)     : 23, 5.9%
## outliers [1]       : 0, 0%
## low counts [2]     : 289, 74%
## (mean count < 41)
## [1] see 'cooksCutoff' argument of ?results
## [2] see 'independentFiltering' argument of ?results
```

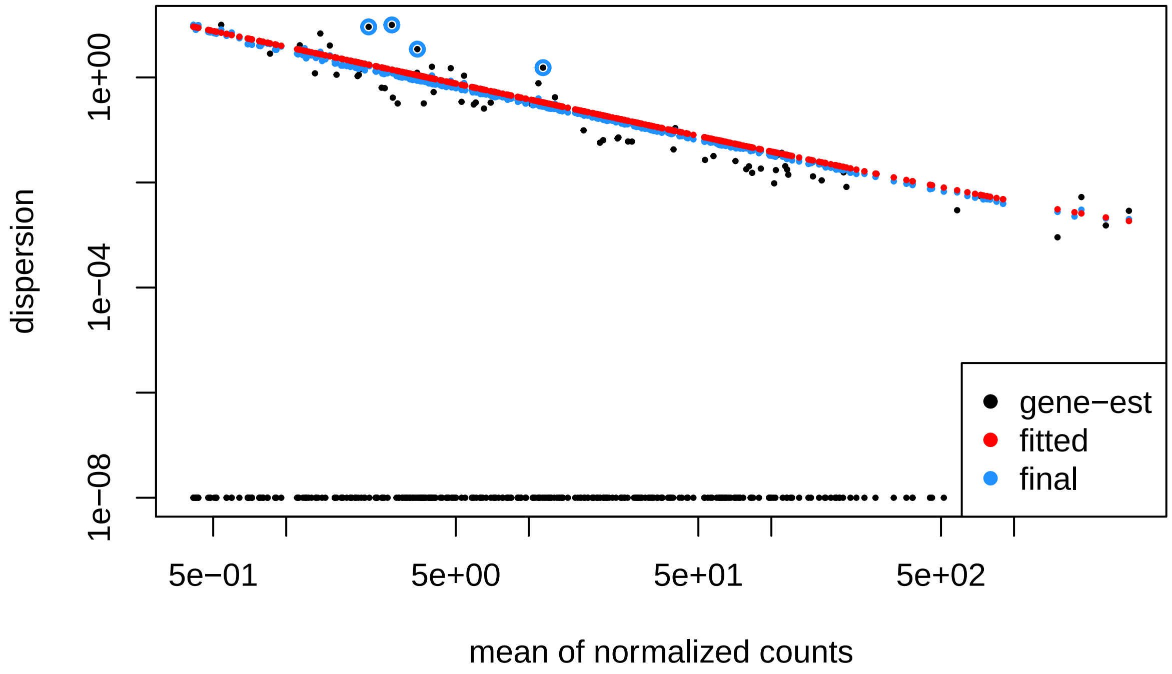
```
sum(res$padj < 0.1, na.rm = TRUE)
```

```
## [1] 34
```

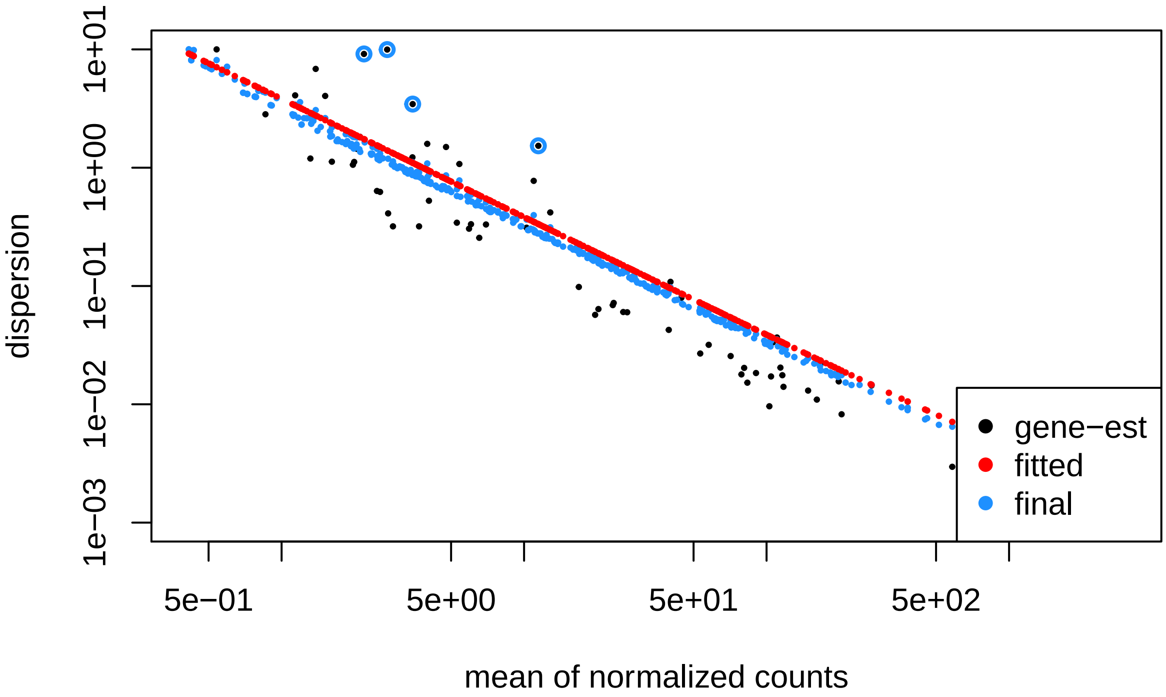
```
#plot MA
plotMA(res, main = "DESeq2", ylim = c(-3, 3))
```



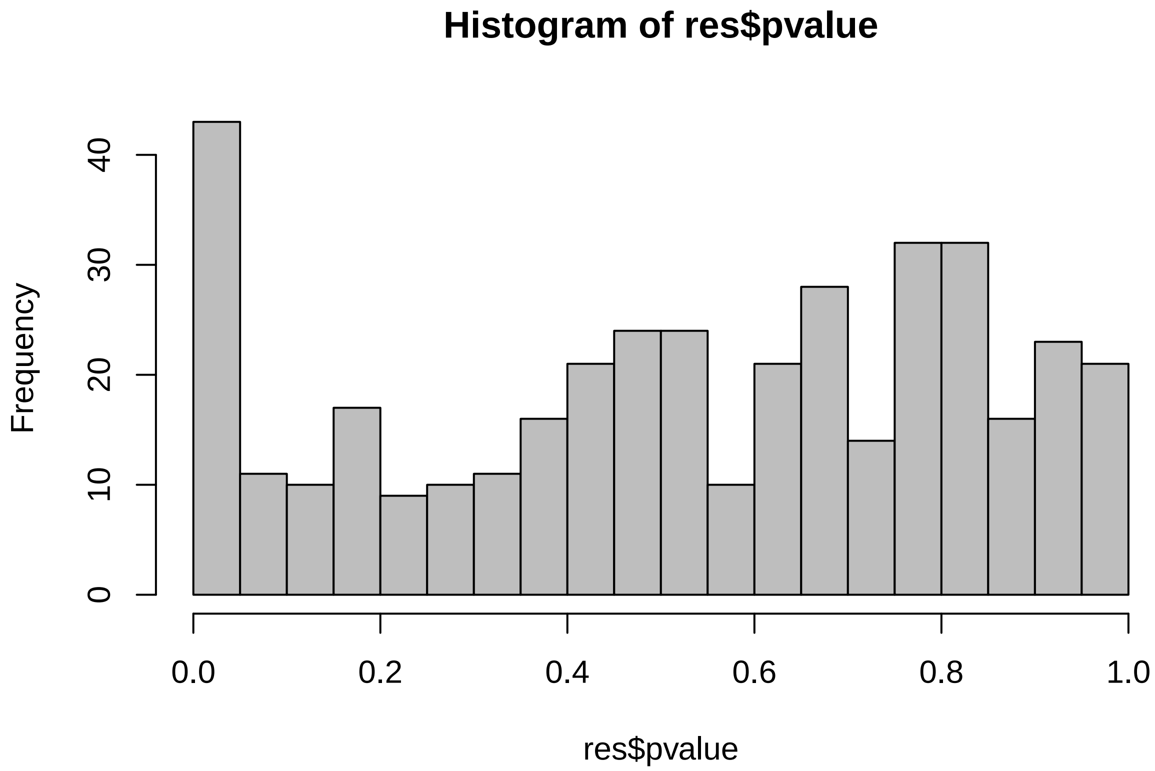
```
#Dispersion plot
plotDispEsts(dds)
```



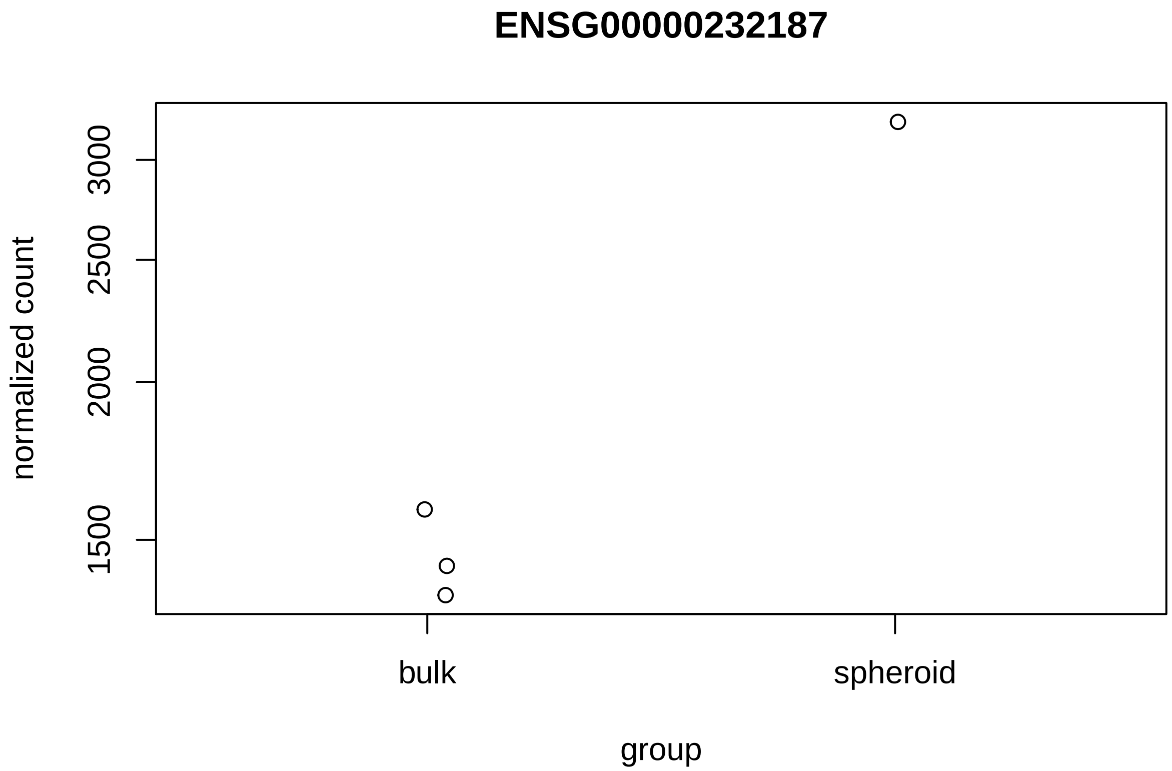
```r
plotDispEsts(dds, ylim = c(1e-3, 1e1))
```



```
#histograma de p-values
hist(res$pvalue, breaks = 20, col = "grey")
```



```
#gene mais significante foi o ENSG00000232187
plotCounts(dds, gene = which.min(res$padj), intgroup = "condition")
```



```
#Exportar resultados para csv
head(as.data.frame(resOrdered))
```

```
##                  baseMean log2FoldChange      lfcSE      stat       pvalue
## ENSG00000232187 1896.2834      1.1413476 0.09813725 11.630116 2.897028e-31
## ENSG00000151846 1776.3687     -0.6463482 0.09256058 -6.982976 2.889925e-12
## ENSG00000180172  728.9480     -0.9835435 0.15044997 -6.537346 6.262005e-11
## ENSG00000198033  773.9874     -0.7931244 0.13797009 -5.748524 9.002590e-09
## ENSG00000223460 2391.1967     -0.4366036 0.08535025 -5.115434 3.130209e-07
## ENSG00000139675  691.1532     -0.6992297 0.14263678 -4.902170 9.478382e-07
##                         padj
## ENSG00000232187 3.012909e-29
## ENSG00000151846 1.502761e-10
## ENSG00000180172 2.170828e-09
## ENSG00000198033 2.340673e-07
## ENSG00000223460 6.510834e-06
## ENSG00000139675 1.642920e-05
```

```
write.csv(as.data.frame(resOrdered), file = "ch13treated.csv")
```

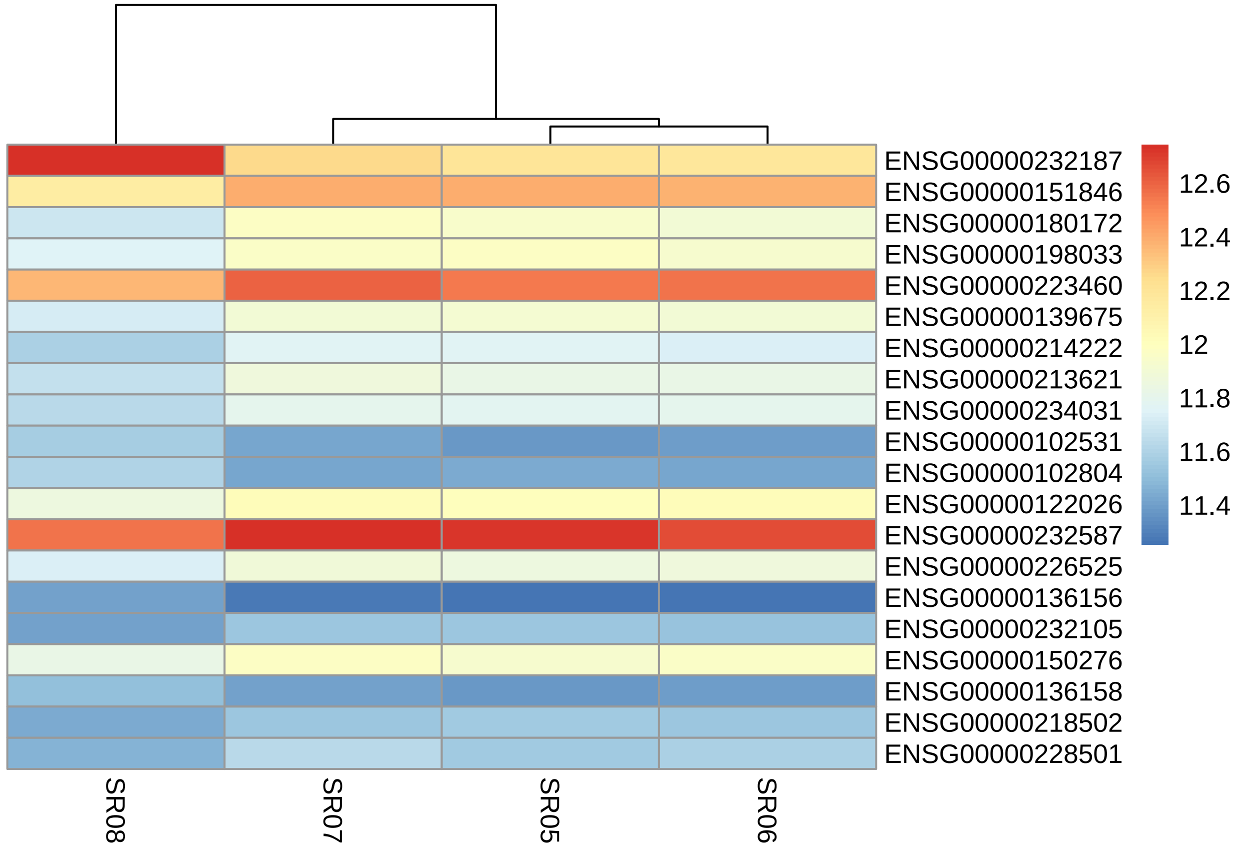
```
#Visualizar os dados
#VST: varianceStabilizingTransformation
vsd <- varianceStabilizingTransformation(dds, blind = FALSE)

#comparar antes e apos normalizar
head(counts(dds), 3)
```

```
##                 SR05 SR06 SR07 SR08
## ENSG00000005810   70   83   66  110
## ENSG00000005812   23   16   16   35
## ENSG00000023516   13   11   12   10
```

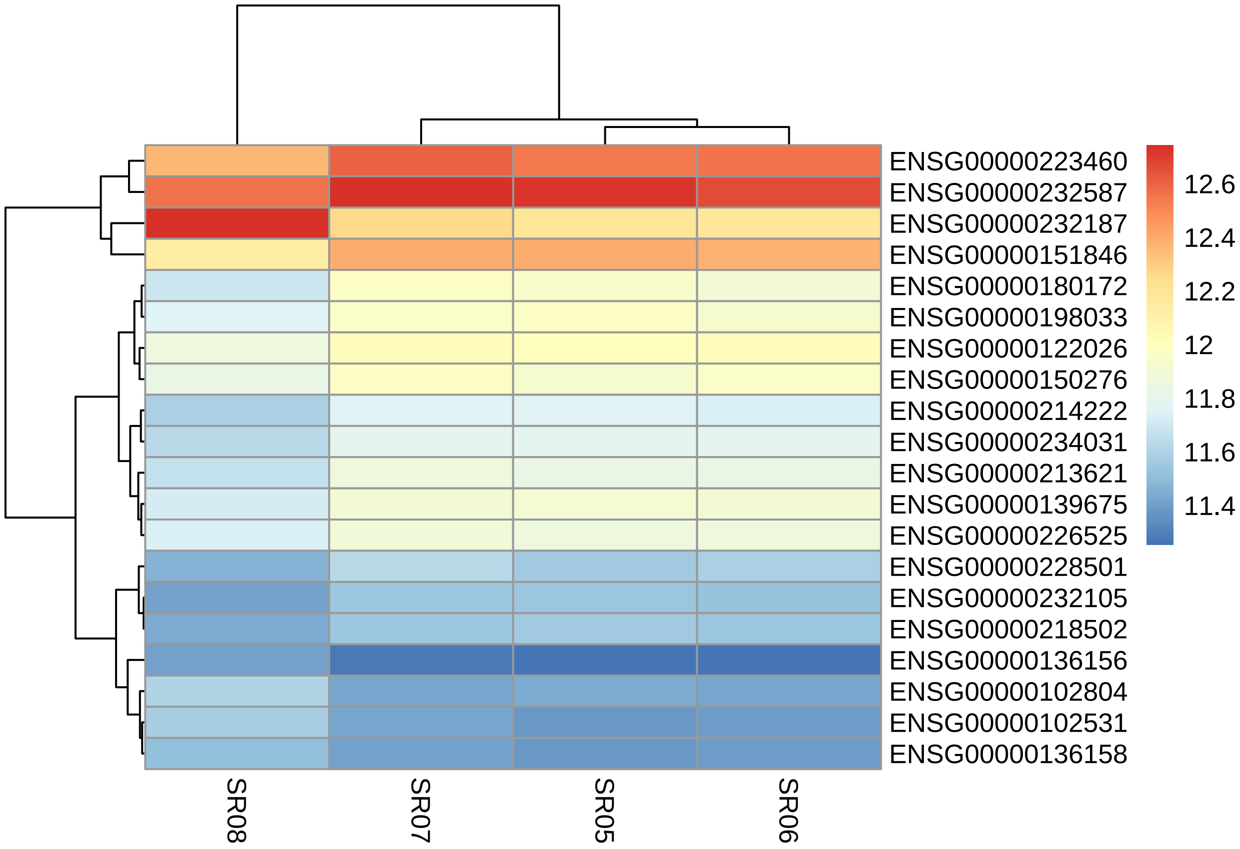
```
head(assay(vsd), 3)
```

```
##                     SR05     SR06     SR07     SR08
## ENSG00000005810 11.32569 11.31933 11.28535 11.41145
## ENSG00000005812 11.20996 11.17072 11.16809 11.25606
## ENSG00000023516 11.17132 11.15082 11.15283 11.16215
```

```
#contruir Heatmap com clustering
select <- rownames(head(resOrdered, 20)) #top 20 apenas
vsd.counts <- assay(vsd)[select,]
df <- as.data.frame(colData(dds)[, c("condition")])

pheatmap(vsd.counts, cluster_rows = FALSE)
```



```
pheatmap(vsd.counts)
```

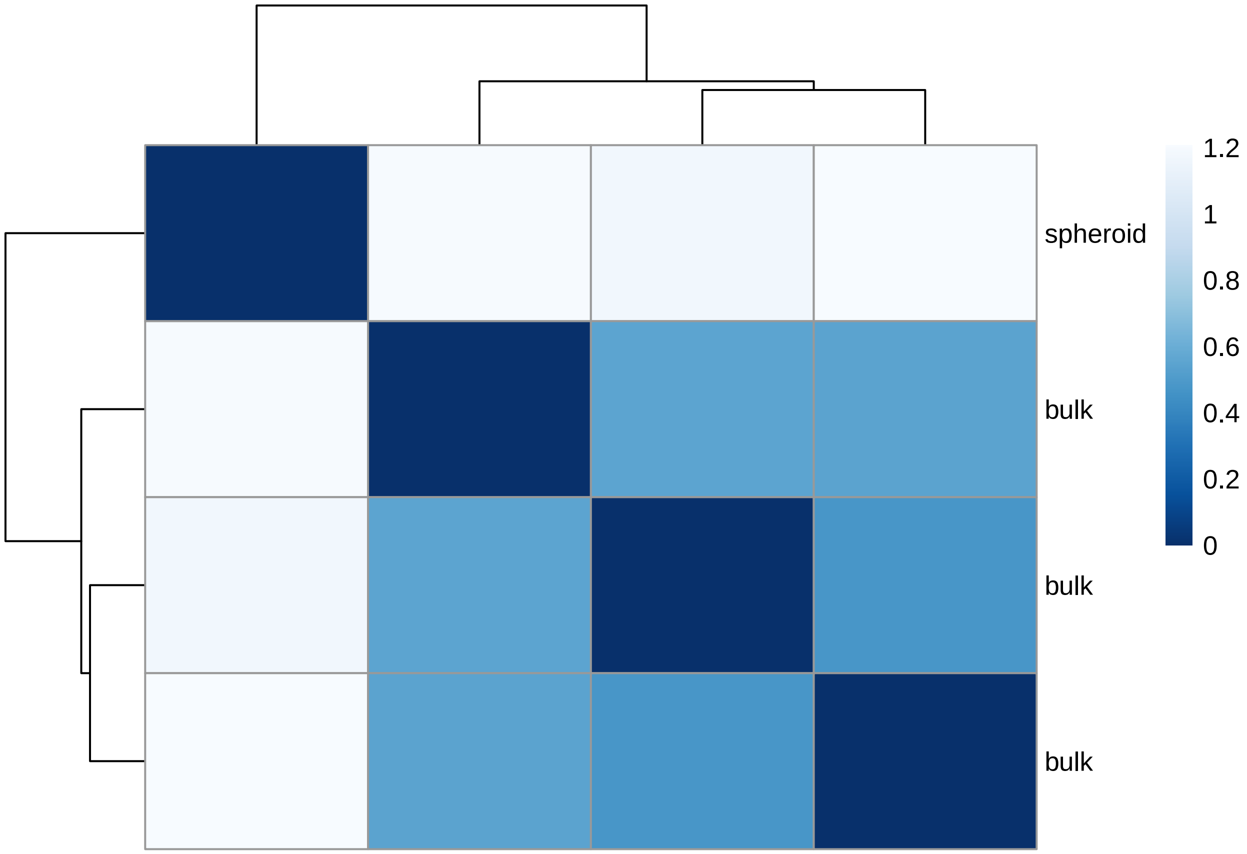


```
#heatmap mostra a diferenca clara entre a expressao genetica entre reads de diferentes fatores
```

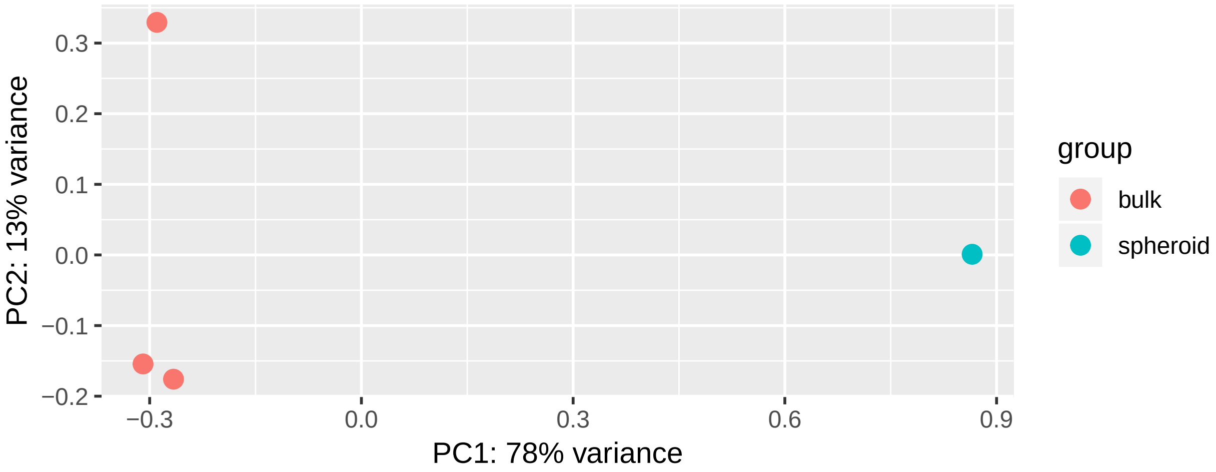
```
#Calculo de distancias de amostras

sampleDists <- dist(t(assay(vsd)))
sampleDistMatrix <- as.matrix(sampleDists)
rownames(sampleDistMatrix) <- dds$condition
colnames(sampleDistMatrix) <- NULL

head(sampleDistMatrix)
```

```
##               [,1]      [,2]      [,3]     [,4]
## bulk     0.0000000 0.5522094 0.5460512 1.201148
## bulk     0.5522094 0.0000000 0.4766472 1.171044
## bulk     0.5460512 0.4766472 0.0000000 1.207249
## spheroid 1.2011478 1.1710440 1.2072488 0.000000
```

```
colors <- colorRampPalette(rev(brewer.pal(9, "Blues")))(255)
#Heatmap com distancias
pheatmap(
  sampleDistMatrix,
  clustering_distance_rows = sampleDists,
  clustering_distance_cols = sampleDists,
  col = colors
)
```



```r
plotPCA(vsd, intgroup = c("condition"))
```


0.3
0.2
0.1
0.0
−0.1
−0.2
PC2: 13% variance
−0.3
0.0
0.3
0.6
0.9
PC1: 78% variance
group
bulk
spheroid