Analise RNASeq a celulas humanas TNBC alinhadas ao cromossoma 17

Grupo 2

14 de Junho 2019

```
#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
```

```
#load de tabela de contagens
sr17 <- read.table("ch17finalreadcount.tab", h = T, row.names = 1)
tail(sr17) # fragmentos do ficheiro que nao sao necessarios para a analise
```

```
##                            SR05    SR06    SR07    SR08
## ENSG00000286272               0       0       0       0
## __no_feature               8463    8782    8748    6234
## __ambiguous                6930    8642    9188    5628
## __too_low_aQual           12021   15125   15213   14224
## __not_aligned           3337046 4080785 4168837 3655715
## __alignment_not_unique        0       0       0       0
```

```
#remover fragmentos
sr17 <- sr17[1:(nrow(sr17) - 5), ]
head(sr17)
```

```
##                 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
```

```
#filtrar genes que nao sao expressos
sr17 <- sr17[rowSums(sr17) > 1,]
head(sr17)
```

```
##                 SR05 SR06 SR07 SR08
## ENSG00000002834   32   36   48   54
## ENSG00000002919    2    2    5    5
## ENSG00000004142   42   51   48   25
## ENSG00000004660    3    5    2    4
## ENSG00000004897   76   74   99   78
## ENSG00000004939    0    1    0    1
```

```
dim(sr17) #dimensoes apos aplicar filtro
```

```
## [1] 907   4
```

```
#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(sr17)
```

```
# Analise Diferencial
dds <- DESeqDataSetFromMatrix(countData = sr17,
                              colData = cd,
                              design = ~ condition)
dds <- DESeq(dds)
```

```
## estimating size factors
```

```
## estimating dispersions
```

```
## gene-wise dispersion estimates
```

```
## mean-dispersion relationship
```

```
## -- note: fitType='parametric', but the dispersion trend was not well captured by the
##    function: y = a/x + b, and a local regression fit was automatically substituted.
##    specify fitType='local' or 'mean' to avoid this message next time.
```

```
## final dispersion estimates
```

```
## fitting model and testing
```

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

```
## log2 fold change (MLE): condition spheroid vs bulk
## Wald test p-value: condition spheroid vs bulk
## DataFrame with 907 rows and 6 columns
##                          baseMean     log2FoldChange             lfcSE
##                         <numeric>          <numeric>         <numeric>
## ENSG00000002834  41.9481151030322  0.662467495786243 0.420323982427059
## ENSG00000002919  3.43835783893967  0.933353878824879  1.63815519087526
## ENSG00000004142   40.307349210972 -0.736497848883214  0.46649272294065
## ENSG00000004660  3.46615362365142   0.43285096607997  1.65036620175181
```

```
## ENSG00000004897  80.2847189835445 0.0845868545734345 0.312641109725426
## ...                          ...                ...                ...
## ENSG00000280136  1.45997919436359   1.85092596543496  3.15474757762427
## ENSG00000280351 0.502529590833507  -1.75138449599058   5.7369608948418
## ENSG00000280852  15.5783377950044 -0.197932041523292 0.732551005907843
## ENSG00000283566  5.08233142145291 -0.293500082065559  1.42013320436944
## ENSG00000284242  151.793697198127  0.459439338675517  0.21603184558129
##                             stat             pvalue               padj
##                        <numeric>          <numeric>          <numeric>
## ENSG00000002834   1.57608778818897  0.115005572973861  0.248752794876796
## ENSG00000002919  0.569759131505845  0.568841078031745                 NA
## ENSG00000004142   -1.5787981519637  0.114382362637681  0.248752794876796
## ENSG00000004660  0.262275709245932  0.793108887913317                 NA
## ENSG00000004897  0.270555764875969  0.786732721711213  0.896602724338549
## ...                          ...                ...                ...
## ENSG00000280136  0.586711272420987  0.557397635788182                 NA
## ENSG00000280351 -0.305280884442717  0.760152233488707                 NA
## ENSG00000280852 -0.270195576727107  0.787009795448766                 NA
## ENSG00000283566 -0.206670811697468  0.836266954133594                 NA
## ENSG00000284242   2.12672042605236 0.0334433193211971 0.0954938445064123
```

```
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
```

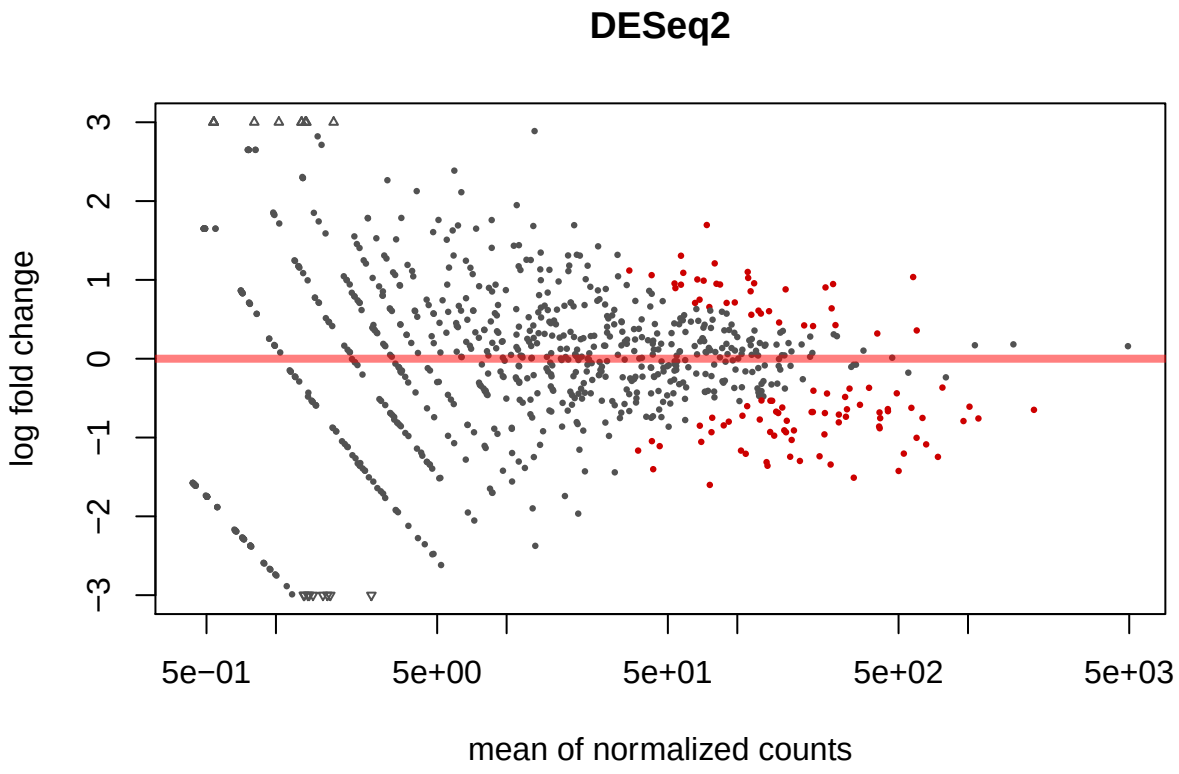
```
resOrdered <- res[order(res$padj),]
summary(res)
```

```
##
## out of 907 with nonzero total read count
## adjusted p-value < 0.1
## LFC > 0 (up)       : 37, 4.1%
## LFC < 0 (down)     : 69, 7.6%
## outliers [1]       : 0, 0%
## low counts [2]     : 615, 68%
## (mean count < 34)
## [1] see 'cooksCutoff' argument of ?results
## [2] see 'independentFiltering' argument of ?results
```

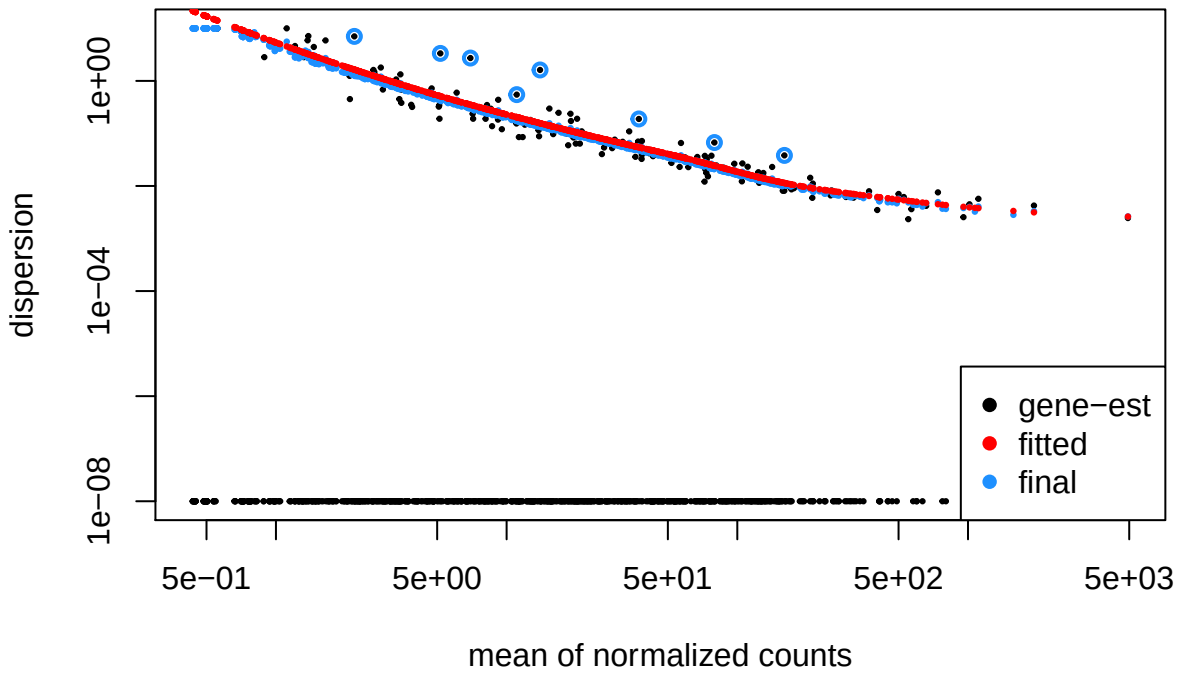
```r
sum(res$padj < 0.1, na.rm = TRUE)
```

```
## [1] 106
```

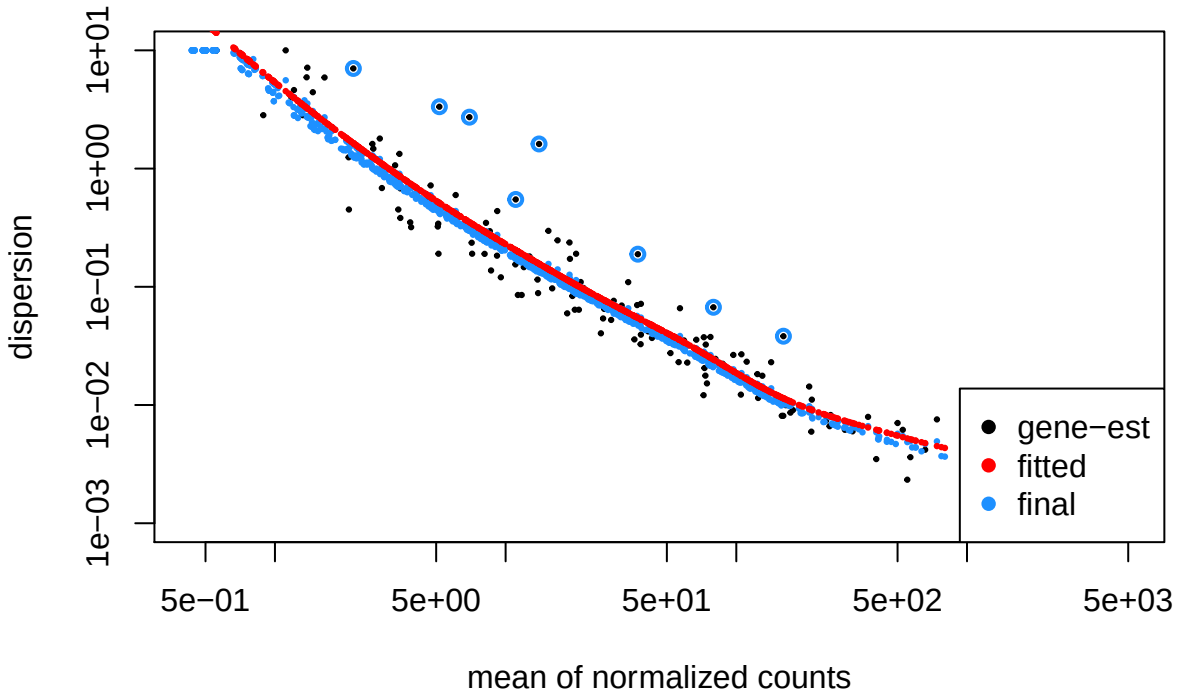
```r
#plot MA
plotMA(res, main = "DESeq2", ylim = c(-3, 3))
```



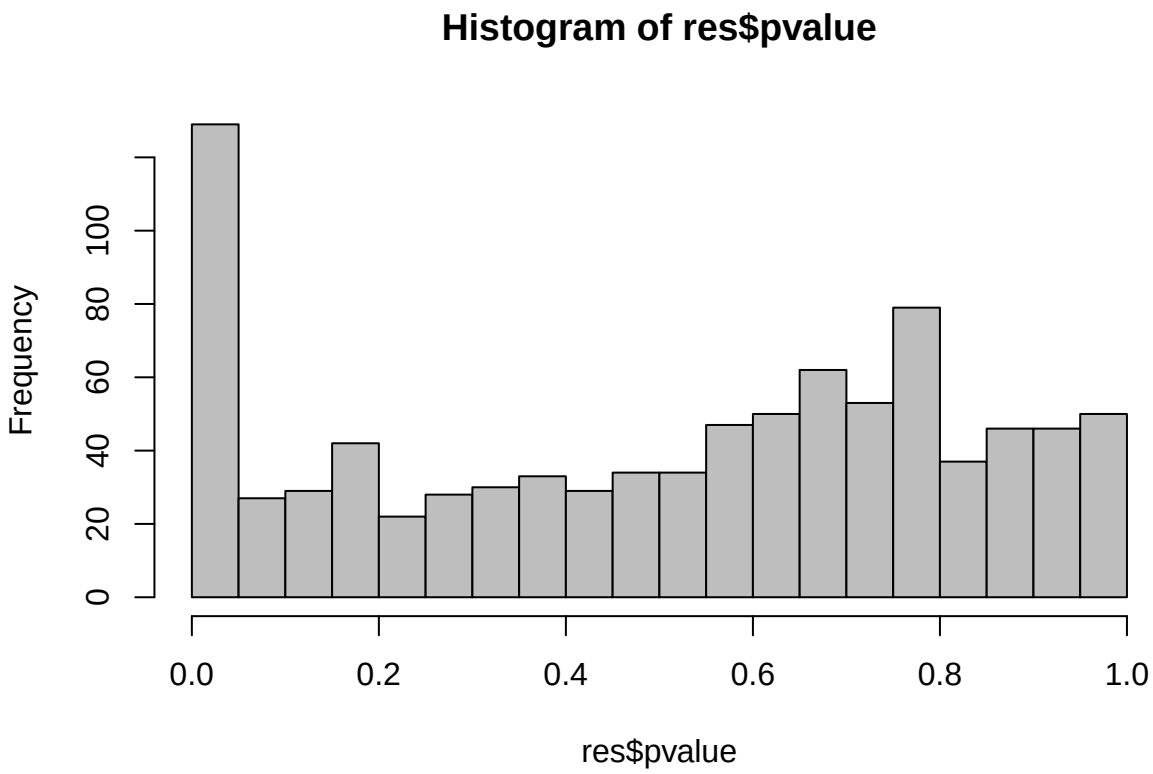

```r
#Dispersion plot
plotDispEsts(dds)
```



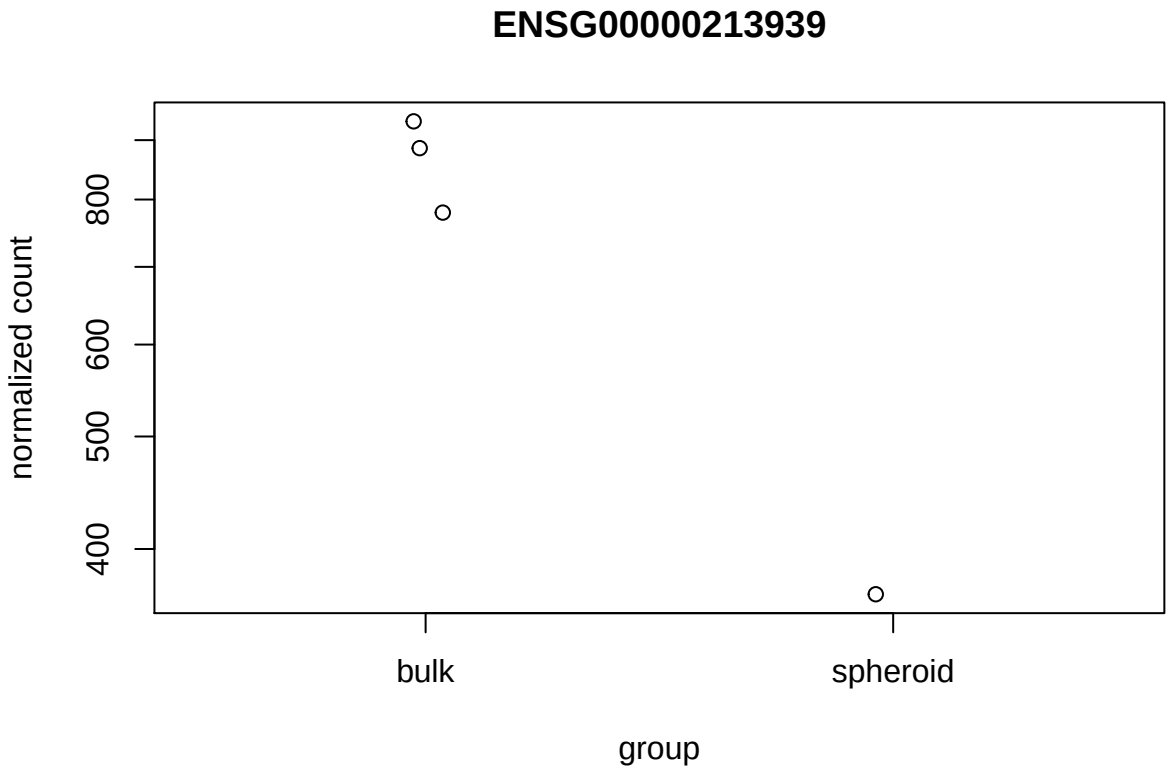

```
plotDispEsts(dds, ylim = c(1e-3, 1e1))
```




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




```
#gene mais significante foi o ENSG00000213939
plotCounts(dds, gene = which.min(res$padj), intgroup = "condition")
```




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

```
##                 baseMean log2FoldChange     lfcSE      stat       pvalue
## ENSG00000213939 740.8129      -1.245596 0.1433853 -8.687050 3.719738e-18
## ENSG00000232344 500.3597      -1.424223 0.1645353 -8.656028 4.884872e-18
## ENSG00000186847 578.4516       1.036271 0.1263921  8.198855 2.426881e-16
## ENSG00000189343 319.7888      -1.509797 0.1910504 -7.902614 2.731150e-15
## ENSG00000215030 526.9668      -1.203947 0.1569213 -7.672300 1.689388e-14
## ENSG00000230897 658.6954      -1.086947 0.1418348 -7.663470 1.809754e-14
##                          padj
## ENSG00000213939 7.131913e-16
## ENSG00000232344 7.131913e-16
## ENSG00000186847 2.362164e-14
## ENSG00000189343 1.993739e-13
## ENSG00000215030 8.807472e-13
## ENSG00000230897 8.807472e-13
```

```
write.csv(as.data.frame(resOrdered), file = "ch17treated.csv")
```

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

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

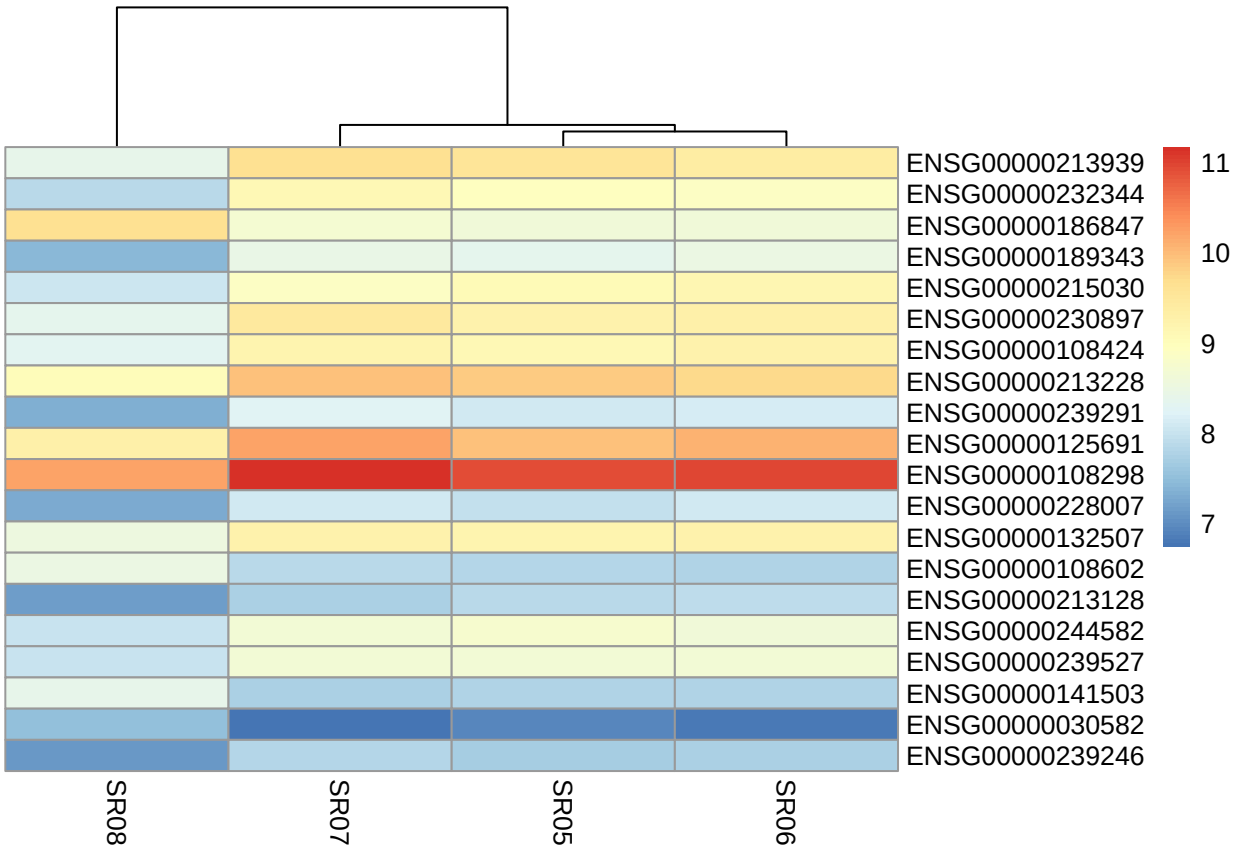
```
##                 SR05 SR06 SR07 SR08
```

```
## ENSG00000002834   32   36   48   54
## ENSG00000002919    2    2    5    5
## ENSG00000004142   42   51   48   25
```

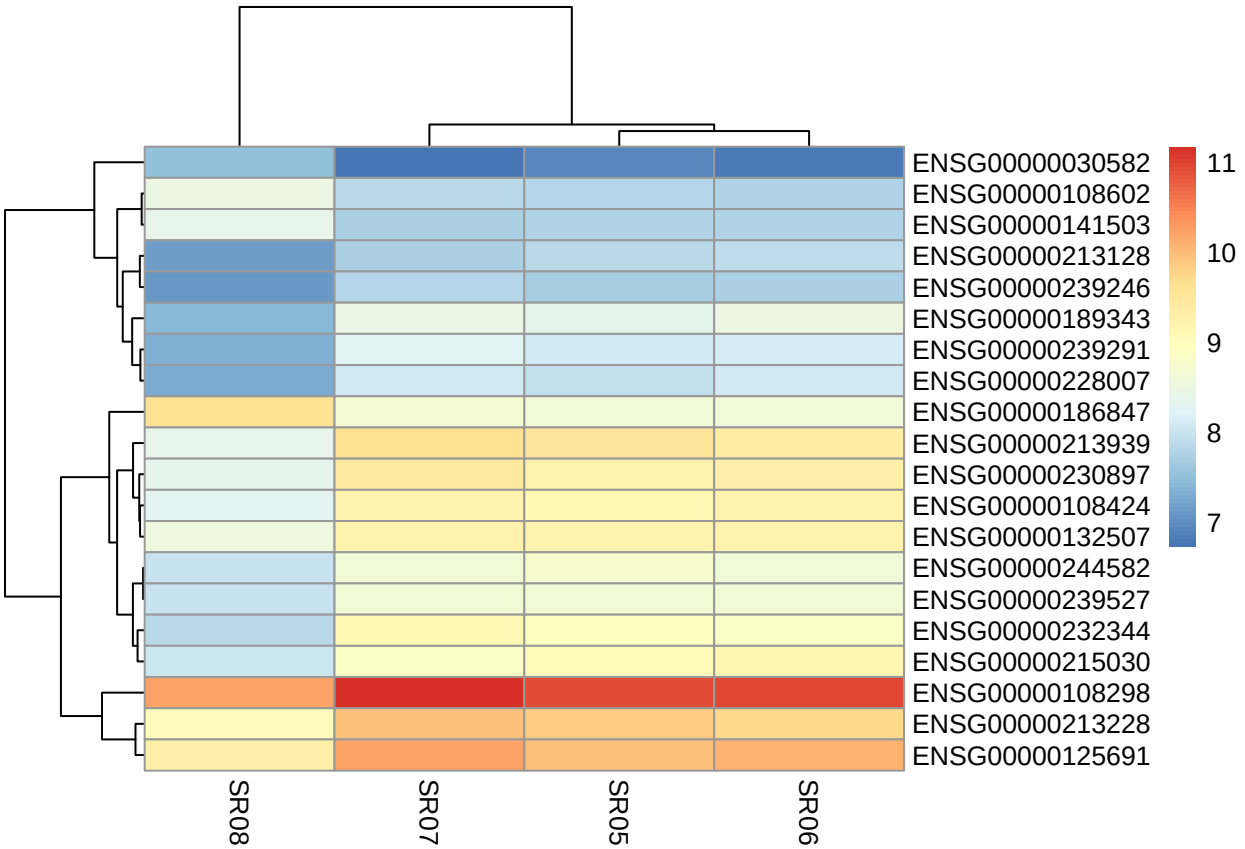
```
head(assay(vsd), 3)
```

```
##                     SR05     SR06     SR07     SR08
## ENSG00000002834 6.704206 6.667723 6.767732 6.912632
## ENSG00000002919 6.161597 6.146047 6.226565 6.252811
## ENSG00000004142 6.814212 6.804407 6.767732 6.604734
```

```
#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)
```

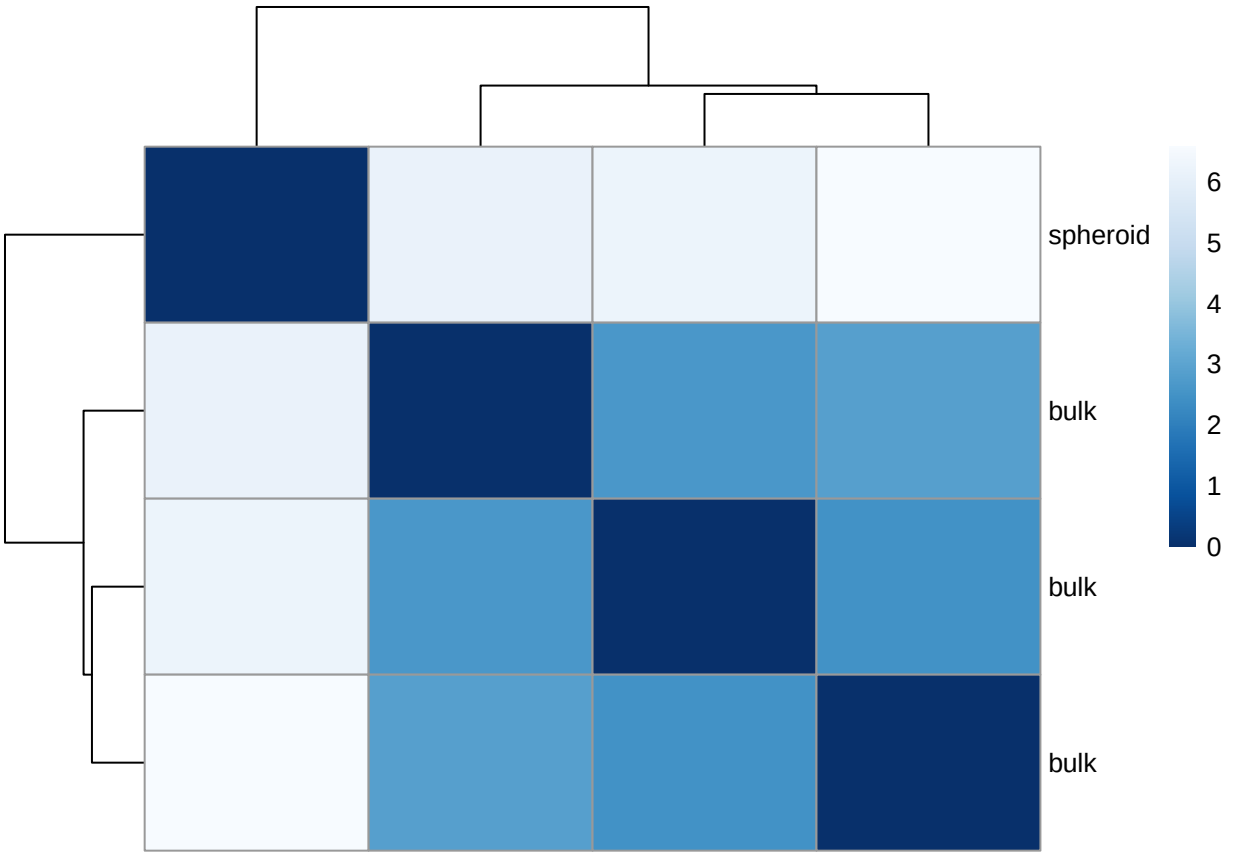



```
#heatmap mostra a diferenca clara entre a expressao genetica entre reads de diferentes fatores
```

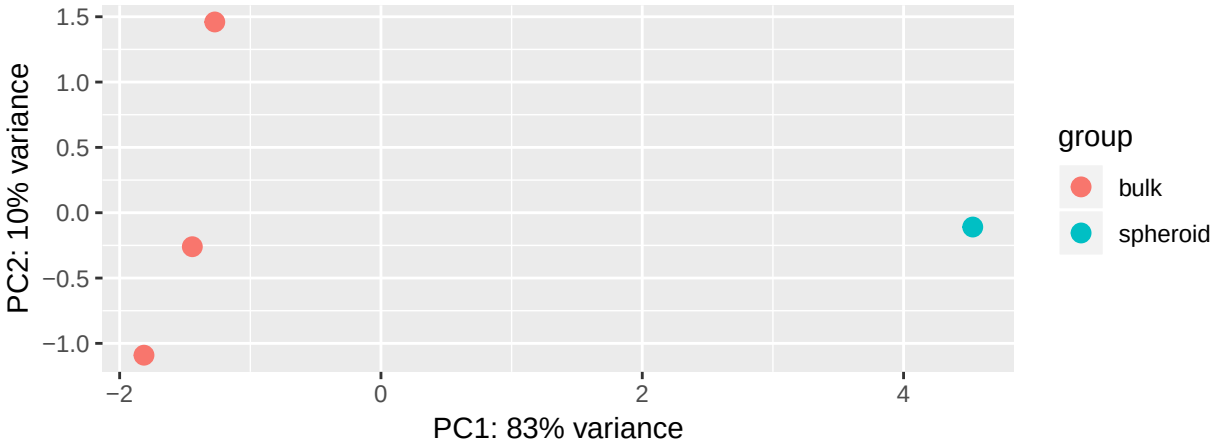
```
#Calculo de distancias de amostras

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

head(sampleDistMatrix)
```

```
##              [,1]     [,2]     [,3]     [,4]
## bulk     0.000000 2.627146 2.871332 6.147208
## bulk     2.627146 0.000000 2.477836 6.226510
## bulk     2.871332 2.477836 0.000000 6.565682
## spheroid 6.147208 6.226510 6.565682 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
)
```




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


1.5
1.0
0.5
0.0
−0.5
−1.0
PC2: 10% variance
−2
0
2
4
PC1: 83% variance
group
bulk
spheroid