Unidad 8: Expresión diferencial

Olga Andrea Hernández Miranda

2024-06-14

Permite comparar los patrones de expresión entre muestras. Se identifican cambios en los perfiles de expresión entre tejidos, etapas, genotipos o tratamientos. Es posible localizar genes que se expresan diferente entre experimentos y conocer las diferencias entre las muestras a través de una prueba de T Se establece un directorio de trabajo

```
directorio <- "C:/Users/andii/OneDrive/Documents/02Fun-R-transcript/data"
setwd(directorio)
```

Calculo de expresion diferencial De un archivo con datos de expresion en TPM se calcula la expresion diferencial entre tratamientos. Se requieren las siguientes librerias:

```
#install.packages("BiocManager")
#BiocManager::install("DESeq2")
library("DESeq2")
```

```
## Loading required package: S4Vectors

## Loading required package: stats4

## Loading required package: BiocGenerics

##
## Attaching package: 'BiocGenerics'

## The following objects are masked from 'package:stats':
##
##     IQR, mad, sd, var, xtabs

## The following objects are masked from 'package:base':
##
##     anyDuplicated, aperm, append, as.data.frame, basename, cbind,
##     colnames, dirname, do.call, duplicated, eval, evalq, Filter, Find,
##     get, grep, grepl, intersect, is.unsorted, lapply, Map, mapply,
##     match, mget, order, paste, pmax, pmax.int, pmin, pmin.int,
##     Position, rank, rbind, Reduce, rownames, sapply, setdiff, sort,
##     table, tapply, union, unique, unsplit, which.max, which.min

##
## Attaching package: 'S4Vectors'
```

```
## The following objects are masked from 'package:base':
##
##     expand.grid, I, unname

## Loading required package: IRanges

##
## Attaching package: 'IRanges'

## The following object is masked from 'package:grDevices':
##
##     windows

## Loading required package: GenomicRanges

## Loading required package: GenomeInfoDb

## Loading required package: SummarizedExperiment

## Loading required package: MatrixGenerics

## Loading required package: matrixStats

##
## Attaching package: 'MatrixGenerics'

## The following objects are masked from 'package:matrixStats':
##
##     colAlls, colAnyNAs, colAnys, colAvgsPerRowSet, colCollapse,
##     colCounts, colCummaxs, colCummins, colCumprods, colCumsums,
##     colDiffs, colIQRDiffs, colIQRs, colLogSumExps, colMadDiffs,
##     colMads, colMaxs, colMeans2, colMedians, colMins, colOrderStats,
##     colProds, colQuantiles, colRanges, colRanks, colSdDiffs, colSds,
##     colSums2, colTabulates, colVarDiffs, colVars, colWeightedMads,
##     colWeightedMeans, colWeightedMedians, colWeightedSds,
##     colWeightedVars, rowAlls, rowAnyNAs, rowAnys, rowAvgsPerColSet,
##     rowCollapse, rowCounts, rowCummaxs, rowCummins, rowCumprods,
##     rowCumsums, rowDiffs, rowIQRDiffs, rowIQRs, rowLogSumExps,
##     rowMadDiffs, rowMads, rowMaxs, rowMeans2, rowMedians, rowMins,
##     rowOrderStats, rowProds, rowQuantiles, rowRanges, rowRanks,
##     rowSdDiffs, rowSds, rowSums2, rowTabulates, rowVarDiffs, rowVars,
##     rowWeightedMads, rowWeightedMeans, rowWeightedMedians,
##     rowWeightedSds, rowWeightedVars

## Loading required package: Biobase

## Welcome to Bioconductor
##
##     Vignettes contain introductory material; view with
##     'browseVignettes()'. To cite Bioconductor, see
##     'citation("Biobase")', and for packages 'citation("pkgname")'.
```

```
##
## Attaching package: 'Biobase'

## The following object is masked from 'package:MatrixGenerics':
##
##     rowMedians

## The following objects are masked from 'package:matrixStats':
##
##     anyMissing, rowMedians
```

Usamos grep para buscar los archivos

```
samplefiles <- grep('conteo', list.files(directorio),value = T)
samplefiles
```

```
## [1] "U8_Etapa1_R1_conteo.txt" "U8_Etapa1_R2_conteo.txt"
## [3] "U8_Etapa2_R1_conteo.txt" "U8_Etapa2_R2_conteo.txt"
```

Asignamos el nombre de la condicionn en simplefiles y ponemos los números al principio para forzar el orden deseado.

```
samplecondition <- c('Etapa1','Etapa1','Etapa2','Etapa2')
```

Creamos un dataframe entre archivos y etiquetas

```
sampletable <- data.frame(sampleName=samplefiles,
                          fileName=samplefiles,
                          condition=samplecondition)

sampletable
```

```
##                sampleName                fileName condition
## 1 U8_Etapa1_R1_conteo.txt U8_Etapa1_R1_conteo.txt    Etapa1
## 2 U8_Etapa1_R2_conteo.txt U8_Etapa1_R2_conteo.txt    Etapa1
## 3 U8_Etapa2_R1_conteo.txt U8_Etapa2_R1_conteo.txt    Etapa2
## 4 U8_Etapa2_R2_conteo.txt U8_Etapa2_R2_conteo.txt    Etapa2
```

Trabajamos con DESeq2 y los datos de conteo

```
ddsHTSeq <- DESeqDataSetFromHTSeqCount(
  sampleTable = sampletable,
  directory = directorio,
  design =~condition
)
```

```
## Warning in DESeqDataSet(se, design = design, ignoreRank): some variables in
## design formula are characters, converting to factors
```

```
ddsHTSeq
```

```
## class: DESeqDataSet
## dim: 57281 4
## metadata(1): version
## assays(1): counts
## rownames(57281): ENSG00000000003.10 ENSG00000000005.5 ...
##   ENSGR0000266731.1 ENSGR0000270726.1
## rowData names(0):
## colnames(4): U8_Etapa1_R1_conteo.txt U8_Etapa1_R2_conteo.txt
##   U8_Etapa2_R1_conteo.txt U8_Etapa2_R2_conteo.txt
## colData names(1): condition
```

Creamos un factor con las etiquetas de las condiciones

```
colData(ddsHTSeq)$condition <- factor(colData(ddsHTSeq)$condition,
                                      levels = c('Etapa2','Etapa1'))
```

Analizamos la expresión diferencial

```
dds <- DESeq(ddsHTSeq)
```

```
## estimating size factors

## estimating dispersions

## gene-wise dispersion estimates

## mean-dispersion relationship

## final dispersion estimates

## fitting model and testing
```

```
res <- results(dds)
res <- res[order(res$padj),]
head(res)
```

```
## log2 fold change (MLE): condition Etapa1 vs Etapa2
## Wald test p-value: condition Etapa1 vs Etapa2
## DataFrame with 6 rows and 6 columns
##                     baseMean log2FoldChange     lfcSE      stat       pvalue
##                    <numeric>      <numeric> <numeric> <numeric>    <numeric>
## ENSG00000001084.6   4243.484       -2.87766 0.0614851  -46.8026  0.00000e+00
## ENSG00000005175.5   3106.371       -3.71974 0.0869161  -42.7968  0.00000e+00
## ENSG00000001036.8   4846.697       -2.00581 0.0596552  -33.6234 7.64080e-248
## ENSG00000000419.8   6010.565        1.12224 0.0503590   22.2849 5.18241e-110
## ENSG00000005189.15   833.348       -2.95639 0.1451688  -20.3652  3.40646e-92
## ENSG00000002726.15   462.353        5.96740 0.3029268   19.6991  2.19280e-86
##                           padj
```

```
##                       <numeric>
## ENSG00000001084.6   0.00000e+00
## ENSG00000005175.5   0.00000e+00
## ENSG00000001036.8  1.09977e-244
## ENSG00000000419.8  5.59442e-107
## ENSG00000005189.15  2.94182e-89
## ENSG00000002726.15  1.57808e-83
```

Resumen de los resultados

```
summary(dds)
```

```
## [1] "DESeqDataSet object of length 57281 with 22 metadata columns"
```

```
#write.csv(res, file = "U8_tabla_ed_crudos.csv")
```

Librerias para volcano plot

```
#install.packages("gplots")
#install.packages("RColorBrewer")
library("gplots")
```

```
##
## Attaching package: 'gplots'

## The following object is masked from 'package:IRanges':
##
##     space

## The following object is masked from 'package:S4Vectors':
##
##     space

## The following object is masked from 'package:stats':
##
##     lowess
```

```
library("RColorBrewer")
```

Valor de corte de p value

```
alpha <- 0.05 #filtrado de los valores alpha
cols <- densCols(res$log2FoldChange,-log10(res$pvalue)) #trabajar con gradiente de color
```

```
## Warning in KernSmooth::bkde2D(x, bandwidth = bandwidth, gridsize = nbin, :
## Binning grid too coarse for current (small) bandwidth: consider increasing
## 'gridsize'
```

```
plot(res$log2FoldChange, -log10(res$padj), col=cols, panel.first=grid(),
     main="Volcano plot", xlab="log2(fold-change)", ylab="-log10(p-value)",
     pch=20, cex=0.6)
```