RNA-seq COVID

Antonio Osamu Katagiri Tanaka - A01212611@itesm.mx

April 17, 2020

D. Blanco-Melo, B. Nilsson-Payant, W.-C. Liu, R. Moeller, M. Panis, D. Sachs, R. Albrecht, B.R. TenOever, SARS-CoV-2 launches a unique transcriptional signature from in vitro, ex vivo, and in vivo systems, BioRxiv. (2020) 2020.03.24.004655. https://doi.org/10.1101/2020.03.24.004655.

```
# Clear all objects (from the workspace)
rm(list = ls())

# Suppress Warning messages
options(warn = -1)

# Turn off scientific notation like 1e+06
options(stringsAsFactors = F)

# INSTALL with:

# if (!requireNamespace("BiocManager", quietly = TRUE))
#     install.packages("BiocManager")
# BiocManager::install("GEOquery")
# BiocManager::install("DESeq2")
# BiocManager::install("limma")

# library(devtools)
# install_github("tpq/exprso")

# LOAD Libs
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------------------------------

## v ggplot2 3.3.0     v purrr   0.3.3
## v tibble  2.1.3     v stringr 1.4.0
## v tidyr   1.0.2     v forcats 0.5.0
## v readr   1.3.1

## -- Conflicts ------------------------------------------------------------------------------------
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(DESeq2)
```

```
## Loading required package: S4Vectors
```

```
## Loading required package: stats4

## Loading required package: BiocGenerics

## Loading required package: parallel

##
## Attaching package: 'BiocGenerics'

## The following objects are masked from 'package:parallel':
##
##     clusterApply, clusterApplyLB, clusterCall, clusterEvalQ,
##     clusterExport, clusterMap, parApply, parCapply, parLapply,
##     parLapplyLB, parRapply, parSapply, parSapplyLB

## The following objects are masked from 'package:dplyr':
##
##     combine, intersect, setdiff, union

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

## The following objects are masked from 'package:base':
##
##     anyDuplicated, 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, which.max, which.min

##
## Attaching package: 'S4Vectors'

## The following object is masked from 'package:tidyr':
##
##     expand

## The following objects are masked from 'package:dplyr':
##
##     first, rename

## The following object is masked from 'package:base':
##
##     expand.grid

## Loading required package: IRanges

##
## Attaching package: 'IRanges'

## The following object is masked from 'package:purrr':
##
##     reduce

## The following objects are masked from 'package:dplyr':
##
##     collapse, desc, slice

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

## Loading required package: GenomicRanges

## Loading required package: GenomeInfoDb

## Loading required package: SummarizedExperiment

## 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")'.

## Loading required package: DelayedArray

## Loading required package: matrixStats

##
## Attaching package: 'matrixStats'

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

## The following object is masked from 'package:dplyr':
##
##     count

## Loading required package: BiocParallel

##
## Attaching package: 'DelayedArray'

## The following objects are masked from 'package:matrixStats':
##
##     colMaxs, colMins, colRanges, rowMaxs, rowMins, rowRanges

## The following object is masked from 'package:purrr':
##
##     simplify

## The following objects are masked from 'package:base':
##
##     aperm, apply, rowsum
library(limma)

##
## Attaching package: 'limma'

## The following object is masked from 'package:DESeq2':
##
##     plotMA

## The following object is masked from 'package:BiocGenerics':
##
##     plotMA
# LOAD provided functions
source("./script_ejercicios.R")

datos <- load("./GSE147507_datos_covid.Rdata")

names(datos_covid)

##  [1] "SARS004_mock_3"          "SARS004_mock_2"
##  [3] "SARS004_mock_1"          "SARS004_CoV2_3"
##  [5] "SARS004_CoV2_2"          "SARS004_CoV2_1"
##  [7] "CoV002.mock3.indexG3"    "CoV002.mock2.indexG2"
##  [9] "CoV002.mock1.indexG1"    "CoV002.CoV2.3.indexG6"
## [11] "CoV002.CoV2.2.indexG5"   "CoV002.CoV2.1.indexG4"
## [13] "svRNA184.mock.3.indexF3" "svRNA184.mock.1.indexF1"
## [15] "svRNA184.RSV.3.indexH9"  "svRNA184.RSV.1.indexF4"
## [17] "X3_9_mock1_13"           "X3_9_mock2_14"
## [19] "X3_9_wt1_15"             "X3_9_wt2_16"
```

```r
dim(datos_covid)
```

```
## [1] 23710    20
```

```r
head(datos_covid)
```

```
##           SARS004_mock_3 SARS004_mock_2 SARS004_mock_1 SARS004_CoV2_3
## DDX11L1                0              0              0              0
## WASH7P                29             24             23             34
## FAM138A                0              0              0              0
## FAM138F                0              0              0              0
## OR4F5                  0              0              0              0
## LOC729737            112            119            113            127
##           SARS004_CoV2_2 SARS004_CoV2_1 CoV002.mock3.indexG3
## DDX11L1                0              0                    0
## WASH7P                19             44                   68
## FAM138A                0              0                    0
## FAM138F                0              0                    0
## OR4F5                  0              0                    0
## LOC729737             84            270                   11
##           CoV002.mock2.indexG2 CoV002.mock1.indexG1 CoV002.CoV2.3.indexG6
## DDX11L1                      0                    0                     0
## WASH7P                      43                   33                    65
## FAM138A                      0                    0                     0
## FAM138F                      0                    0                     0
## OR4F5                        0                    0                     0
## LOC729737                    3                    6                     8
##           CoV002.CoV2.2.indexG5 CoV002.CoV2.1.indexG4 svRNA184.mock.3.indexF3
## DDX11L1                       1                     1                      1
## WASH7P                       79                    48                    184
## FAM138A                       0                     0                      0
## FAM138F                       0                     0                      0
## OR4F5                         0                     0                      0
## LOC729737                    10                    10                    108
##           svRNA184.mock.1.indexF1 svRNA184.RSV.3.indexH9 svRNA184.RSV.1.indexF4
## DDX11L1                         0                     0                      0
## WASH7P                        128                    51                     43
## FAM138A                         0                     0                      0
## FAM138F                         0                     0                      0
## OR4F5                           0                     0                      0
## LOC729737                      95                    37                     11
##           X3_9_mock1_13 X3_9_mock2_14 X3_9_wt1_15 X3_9_wt2_16
## DDX11L1               0             0           0           0
## WASH7P               15            12           3           3
## FAM138A               0             0           0           0
## FAM138F               0             0           0           0
## OR4F5                 0             0           0           0
## LOC729737             1             5           0           2
```

Cell Culture: * Normal human bronchial epithelial (NHBE) * Human adenocarcinomic alveolar basal epithelial (A549) cells

Viruses: * SARS-related coronavirus 2 (SARS.CoV.2) * influenza A/Puerto Rico/8/1934 (H1N1) virus (IAV) * human respiratory syncytial virus (RSV)

Filtrar genes con baja expresión y con bajos conteos

```r
# FILTRAR genes - conteos

# Para cada gen, contar el número de muestras con mayor a 5 conteos
datos_covid_filter <-
    apply(datos_covid, 1, function(x)
```

```
        length(which(x >= 5)))
table(datos_covid_filter)
```

```
## datos_covid_filter
##     0     1     2     3     4     5     6     7     8     9    10    11    12
##  6527   668   380   307   262   238   655   240   221   205   250   240   245
##    13    14    15    16    17    18    19    20
##   273   445   282   261   289   370   575 10777
```

```
# Filtrar genes con menor a 2 muestras con más de 5 conteos
datos_covid <- datos_covid[which(datos_covid_filter >= 2),]

# Let's take a LOOK
names(datos_covid)
```

```
##  [1] "SARS004_mock_3"          "SARS004_mock_2"
##  [3] "SARS004_mock_1"          "SARS004_CoV2_3"
##  [5] "SARS004_CoV2_2"          "SARS004_CoV2_1"
##  [7] "CoV002.mock3.indexG3"    "CoV002.mock2.indexG2"
##  [9] "CoV002.mock1.indexG1"    "CoV002.CoV2.3.indexG6"
## [11] "CoV002.CoV2.2.indexG5"   "CoV002.CoV2.1.indexG4"
## [13] "svRNA184.mock.3.indexF3" "svRNA184.mock.1.indexF1"
## [15] "svRNA184.RSV.3.indexH9"  "svRNA184.RSV.1.indexF4"
## [17] "X3_9_mock1_13"           "X3_9_mock2_14"
## [19] "X3_9_wt1_15"             "X3_9_wt2_16"
```

```
dim(datos_covid)
```

```
## [1] 16515    20
```

```
head(datos_covid)
```

```
##              SARS004_mock_3 SARS004_mock_2 SARS004_mock_1 SARS004_CoV2_3
## WASH7P                   29             24             23             34
## LOC729737               112            119            113            127
## LOC100133331             19             21             36             16
## LOC100288069             13             17             27             19
## LINC00115                 9             12             26             12
## LOC643837                93             74            143             88
##              SARS004_CoV2_2 SARS004_CoV2_1 CoV002.mock3.indexG3
## WASH7P                   19             44                   68
## LOC729737                84            270                   11
## LOC100133331             23             54                   54
## LOC100288069              9             50                   23
## LINC00115                 7             28                   29
## LOC643837                95            301                  250
##              CoV002.mock2.indexG2 CoV002.mock1.indexG1 CoV002.CoV2.3.indexG6
## WASH7P                         43                   33                   65
## LOC729737                       3                    6                    8
## LOC100133331                   23                   20                   45
## LOC100288069                   18                    5                   30
## LINC00115                      15                    2                   17
## LOC643837                     127                   83                  243
##              CoV002.CoV2.2.indexG5 CoV002.CoV2.1.indexG4
## WASH7P                          79                   48
## LOC729737                       10                   10
## LOC100133331                    42                   36
## LOC100288069                    17                   14
## LINC00115                       13                   16
## LOC643837                      197                  136
##              svRNA184.mock.3.indexF3 svRNA184.mock.1.indexF1
## WASH7P                           184                     128
## LOC729737                        108                      95
## LOC100133331                     127                      93
```

```
## LOC100288069                      7                     12
## LINC00115                        38                     24
## LOC643837                       145                     95
##              svRNA184.RSV.3.indexH9 svRNA184.RSV.1.indexF4 X3_9_mock1_13
## WASH7P                           51                     43            15
## LOC729737                        37                     11             1
## LOC100133331                     25                     16            16
## LOC100288069                      4                      3             5
## LINC00115                        13                      6             5
## LOC643837                        41                     30            39
##              X3_9_mock2_14 X3_9_wt1_15 X3_9_wt2_16
## WASH7P                  12           3           3
## LOC729737                5           0           2
## LOC100133331            14          10           3
## LOC100288069            15           8           3
## LINC00115               11           4           0
## LOC643837               81          77          18
```

Classify samples

SARS004_mock: “SARS004_mock_3” “SARS004_mock_2” “SARS004_mock_1”

SARS004_CoV2: “SARS004_CoV2_3” “SARS004_CoV2_2” “SARS004_CoV2_1”

CoV002.mock: “CoV002.mock3.indexG3” “CoV002.mock2.indexG2” “CoV002.mock1.indexG1”

CoV002.CoV2: “CoV002.CoV2.3.indexG6” “CoV002.CoV2.2.indexG5” “CoV002.CoV2.1.indexG4”

svRNA184.mock: “svRNA184.mock.3.indexF3” “svRNA184.mock.1.indexF1”

svRNA184.RSV: “svRNA184.RSV.3.indexH9” “svRNA184.RSV.1.indexF4”

X3_9_mock: “X3_9_mock1_13” “X3_9_mock2_14”

X3_9_wt: “X3_9_wt1_15” “X3_9_wt2_16”

```
SARS004_mock  = datos_covid[,grep('SARS004_mock',  colnames(datos_covid))]
SARS004_CoV2  = datos_covid[,grep('SARS004_CoV2',  colnames(datos_covid))]

CoV002.mock   = datos_covid[,grep('CoV002.mock',   colnames(datos_covid))]
CoV002.CoV2   = datos_covid[,grep('CoV002.CoV2',   colnames(datos_covid))]

svRNA184.mock = datos_covid[,grep('svRNA184.mock', colnames(datos_covid))]
svRNA184.RSV  = datos_covid[,grep('svRNA184.RSV',  colnames(datos_covid))]

X3_9_mock     = datos_covid[,grep('X3_9_mock',     colnames(datos_covid))]
X3_9_wt       = datos_covid[,grep('X3_9_wt',       colnames(datos_covid))]
```

Apply DESeq2 to SARS004 (SARS)

```
tbl_merge = merge(SARS004_mock, SARS004_CoV2, by="row.names", all=TRUE)
rownames(tbl_merge) <- tbl_merge[,1] # First column contains the row names
tbl_merge = tbl_merge %>% select(-contains(c("Row.names"))) # DROP extra data
print(head(tbl_merge))
```

```
##          SARS004_mock_3 SARS004_mock_2 SARS004_mock_1 SARS004_CoV2_3
## A1BG                  8              7              6              7
## A1BG-AS1             23             34             33             28
## A1CF                  0              0              0              0
## A2M                   2              2              0              1
## A2M-AS1               3              3              0              1
## A2ML1              1487           1267           2156           1259
##          SARS004_CoV2_2 SARS004_CoV2_1
## A1BG                  3             22
```

```
## A1BG-AS1             18             68
## A1CF                  0              0
## A2M                   0              8
## A2M-AS1               0             12
## A2ML1              1280           5239
```

```
# Conteos
aux_classes = rep(0, times = ncol(tbl_merge)) # CLASSIFY infected samples as 0s
aux_classes[grep(pattern = "mock", x = colnames(tbl_merge))] = 1 # and mocks as 1s
aux_classes
```

```
## [1] 1 1 1 0 0 0
```

```
colnames(tbl_merge)
```

```
## [1] "SARS004_mock_3" "SARS004_mock_2" "SARS004_mock_1" "SARS004_CoV2_3"
## [5] "SARS004_CoV2_2" "SARS004_CoV2_1"
```

```
count_results_SARS004 = DESeq_func(matrix_c = tbl_merge, classes_c = aux_classes)
```

```
## estimating size factors
```

```
## estimating dispersions
```

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

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

```
## final dispersion estimates
```

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

```
count_results_SARS004 = count_results_SARS004[order(count_results_SARS004$pvalue),]
summary(count_results_SARS004)
```

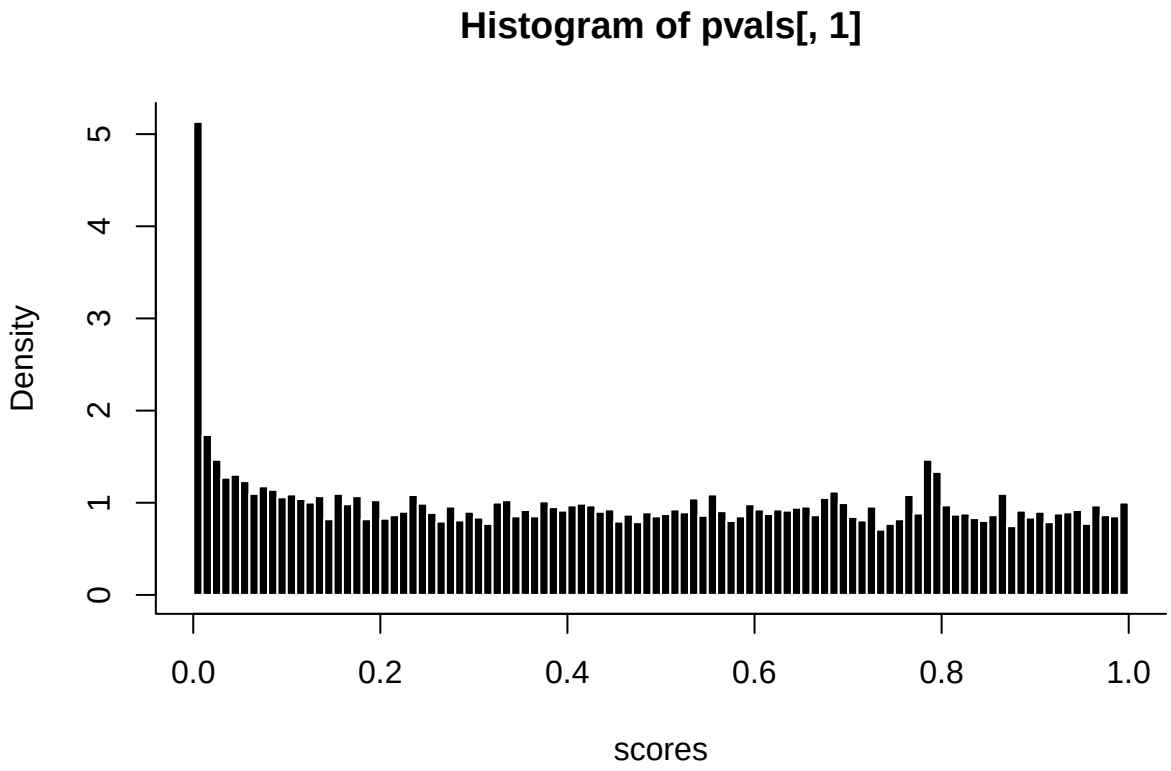
```
##     baseMean          log2FoldChange        lfcSE              stat
##  Min.   :     0.00   Min.   :-5.0817   Min.   :0.0771   Min.   :-22.0089
##  1st Qu.:    17.44   1st Qu.:-0.1576   1st Qu.:0.1252   1st Qu.: -0.7192
##  Median :   181.43   Median : 0.0063   Median :0.1829   Median :  0.0308
##  Mean   :   832.79   Mean   : 0.0146   Mean   :0.4908   Mean   : -0.0437
##  3rd Qu.:   663.88   3rd Qu.: 0.1909   3rd Qu.:0.4475   3rd Qu.:  0.7730
##  Max.   :135168.60   Max.   : 4.5330   Max.   :4.0805   Max.   :  8.0613
##                      NA's   :561       NA's   :561      NA's   :561
##      pvalue            padj
##  Min.   :0.0000   Min.   :0.000
##  1st Qu.:0.1823   1st Qu.:0.569
##  Median :0.4558   Median :0.804
##  Mean   :0.4615   Mean   :0.712
##  3rd Qu.:0.7277   3rd Qu.:0.916
##  Max.   :1.0000   Max.   :1.000
##  NA's   :561      NA's   :3957
```

```
head(count_results_SARS004)
```

```
##         baseMean log2FoldChange     lfcSE      stat        pvalue          padj
## IL8    2374.7575      -2.335177 0.1061013 -22.00894 2.364807e-107 2.969725e-103
## CCL20   412.3543      -3.146069 0.1649441 -19.07355  4.188415e-81  2.629906e-77
## SAA2    576.0496      -2.424539 0.1282341 -18.90713  9.963962e-80  4.170914e-76
## SAA1   3317.5564      -2.223965 0.1396037 -15.93056  3.888422e-57  1.220770e-53
## IL36G   271.4601      -2.734517 0.1761112 -15.52722  2.270245e-54  5.701948e-51
## S100A8 1707.6257      -1.871364 0.1222721 -15.30492  7.089426e-53  1.483817e-49
```

```
# HISTOGRAM
pvals = count_results_SARS004["pvalue"]
hist(
    pvals[, 1],
    prob = TRUE,
    col = "black",
    border = "white",
```

```
    xlab = "scores",
    breaks = 100
)
box(bty = "l")
# Draw density function (assuming normal dist)
score_mean = mean(pvals[, 1])
score_sd   = sd(pvals[, 1])
curve(
    dnorm(x, mean = score_mean, sd = score_sd),
    add = TRUE,
    col = "red",
    lwd = 2
)
```




```
# Let's take a look to some genes
# count_mat_28v0["ENSG00000128567.16_PODXL",]
# count_mat_28v0["ENSG00000185559.14_DLK1",]
```

Apply DESeq2 to CoV002 (SARS-related coronavirus 2)

```
tbl_merge = merge(CoV002.mock, CoV002.CoV2, by="row.names", all=TRUE)
rownames(tbl_merge) <- tbl_merge[,1] # First column contains the row names
tbl_merge = tbl_merge %>% select(-contains(c("Row.names"))) # DROP extra data
print(head(tbl_merge))
```

```
##          CoV002.mock3.indexG3 CoV002.mock2.indexG2 CoV002.mock1.indexG1
## A1BG                      101                   47                   48
## A1BG-AS1                   16                    6                    4
## A1CF                        4                    5                    3
## A2M                         0                    1                    0
## A2M-AS1                    37                   15                   12
## A2ML1                       1                    0                    0
##          CoV002.CoV2.3.indexG6 CoV002.CoV2.2.indexG5 CoV002.CoV2.1.indexG4
```

```
## A1BG                     133                  129                   91
## A1BG-AS1                  27                   19                   19
## A1CF                       8                    9                    5
## A2M                        0                    0                    0
## A2M-AS1                   47                   32                   17
## A2ML1                      0                    0                    1
```

```
# Conteos
aux_classes = rep(0, times = ncol(tbl_merge)) # CLASSIFY infected samples as 0s
aux_classes[grep(pattern = "mock", x = colnames(tbl_merge))] = 1 # and mocks as 1s
aux_classes
```

```
## [1] 1 1 1 0 0 0
```

```
colnames(tbl_merge)
```

```
## [1] "CoV002.mock3.indexG3"  "CoV002.mock2.indexG2"  "CoV002.mock1.indexG1"
## [4] "CoV002.CoV2.3.indexG6" "CoV002.CoV2.2.indexG5" "CoV002.CoV2.1.indexG4"
```

```
count_results_CoV002 = DESeq_func(matrix_c = tbl_merge, classes_c = aux_classes)
```

```
## estimating size factors

## estimating dispersions

## gene-wise dispersion estimates

## mean-dispersion relationship

## final dispersion estimates

## fitting model and testing
```

```
count_results_CoV002 = count_results_CoV002[order(count_results_CoV002$pvalue),]
summary(count_results_CoV002)
```

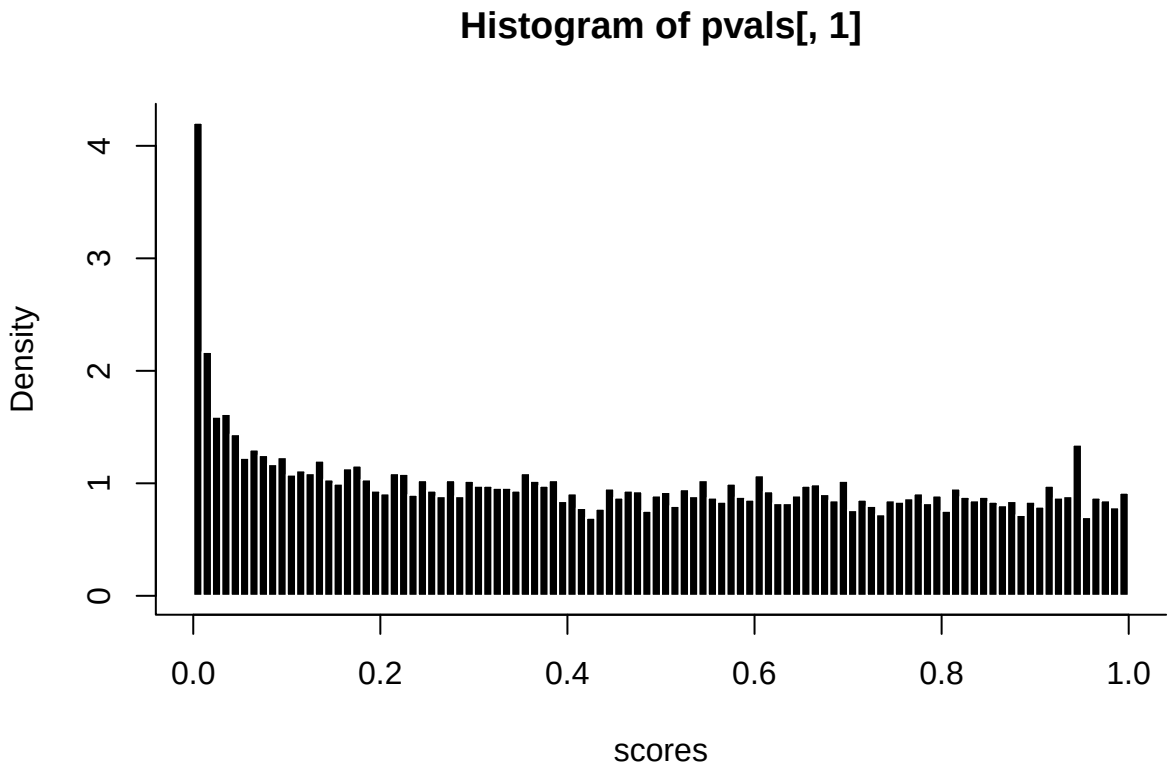
```
##     baseMean          log2FoldChange        lfcSE              stat
##  Min.   :     0.00   Min.   :-6.0860   Min.   :0.0724   Min.   :-34.3980
##  1st Qu.:    24.97   1st Qu.:-0.1699   1st Qu.:0.1087   1st Qu.: -0.8638
##  Median :   273.65   Median :-0.0065   Median :0.1458   Median : -0.0330
##  Mean   :   946.00   Mean   :-0.0092   Mean   :0.3873   Mean   : -0.1118
##  3rd Qu.:   844.23   3rd Qu.: 0.1344   3rd Qu.:0.3426   3rd Qu.:  0.7286
##  Max.   :145716.11   Max.   : 4.0619   Max.   :4.0805   Max.   :  6.5449
##                      NA's   :370       NA's   :370      NA's   :370
##      pvalue             padj
##  Min.   :0.0000   Min.   :0.0000
##  1st Qu.:0.1715   1st Qu.:0.5664
##  Median :0.4314   Median :0.7605
##  Mean   :0.4485   Mean   :0.7036
##  3rd Qu.:0.7088   3rd Qu.:0.8972
##  Max.   :1.0000   Max.   :1.0000
##  NA's   :370      NA's   :2871
```

```
head(count_results_CoV002)
```

```
##        baseMean log2FoldChange     lfcSE      stat        pvalue          padj
## IFI6  1308.9324      -4.333094 0.1259693 -34.39801 2.700377e-259 3.684394e-255
## IFIT1  590.6098      -4.296126 0.1695301 -25.34138 1.118549e-141 7.630738e-138
## ISG15  486.2893      -3.766134 0.1540151 -24.45302 4.673998e-132 2.125734e-128
## MX1    386.1696      -5.066631 0.2275617 -22.26487 8.095413e-110 2.761345e-106
## IRF9   761.8900      -2.159944 0.1103179 -19.57927  2.323191e-85  6.339523e-82
## IRF7   357.9020      -3.184034 0.1721120 -18.49978  2.073161e-76  4.714369e-73
```

```
# HISTOGRAM
pvals = count_results_CoV002["pvalue"]
hist(
    pvals[, 1],
    prob = TRUE,
    col = "black",
```

```
    border = "white",
    xlab = "scores",
    breaks = 100
)
box(bty = "l")
# Draw density function (assuming normal dist)
score_mean = mean(pvals[, 1])
score_sd   = sd(pvals[, 1])
curve(
    dnorm(x, mean = score_mean, sd = score_sd),
    add = TRUE,
    col = "red",
    lwd = 2
)
```




```
# Let's take a look to some genes
# count_mat_28v0["ENSG00000128567.16_PODXL",]
# count_mat_28v0["ENSG00000185559.14_DLK1",]
```

Apply DESeq2 to svRNA184 (human respiratory syncytial virus)

```
tbl_merge = merge(svRNA184.mock, svRNA184.RSV, by="row.names", all=TRUE)
rownames(tbl_merge) <- tbl_merge[,1] # First column contains the row names
tbl_merge = tbl_merge %>% select(-contains(c("Row.names"))) # DROP extra data
print(head(tbl_merge))
```

```
##          svRNA184.mock.3.indexF3 svRNA184.mock.1.indexF1 svRNA184.RSV.3.indexH9
## A1BG                         422                     257                    158
## A1BG-AS1                      43                      26                      5
## A1CF                         127                      75                     16
## A2M                            1                       6                      3
## A2M-AS1                       36                      40                      6
## A2ML1                          0                       0                      0
```

```
##          svRNA184.RSV.1.indexF4
## A1BG                         82
## A1BG-AS1                      3
## A1CF                          9
## A2M                           1
## A2M-AS1                       4
## A2ML1                         0
```

```
# Conteos
aux_classes = rep(0, times = ncol(tbl_merge)) # CLASSIFY infected samples as 0s
aux_classes[grep(pattern = "mock", x = colnames(tbl_merge))] = 1 # and mocks as 1s
aux_classes
```

```
## [1] 1 1 0 0
```

```
colnames(tbl_merge)
```

```
## [1] "svRNA184.mock.3.indexF3" "svRNA184.mock.1.indexF1"
## [3] "svRNA184.RSV.3.indexH9"  "svRNA184.RSV.1.indexF4"
```

```
count_results_svRNA184 = DESeq_func(matrix_c = tbl_merge, classes_c = aux_classes)
```

```
## estimating size factors
```

```
## estimating dispersions
```

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

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

```
## final dispersion estimates
```

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

```
count_results_svRNA184 = count_results_svRNA184[order(count_results_svRNA184$pvalue),]
summary(count_results_svRNA184)
```

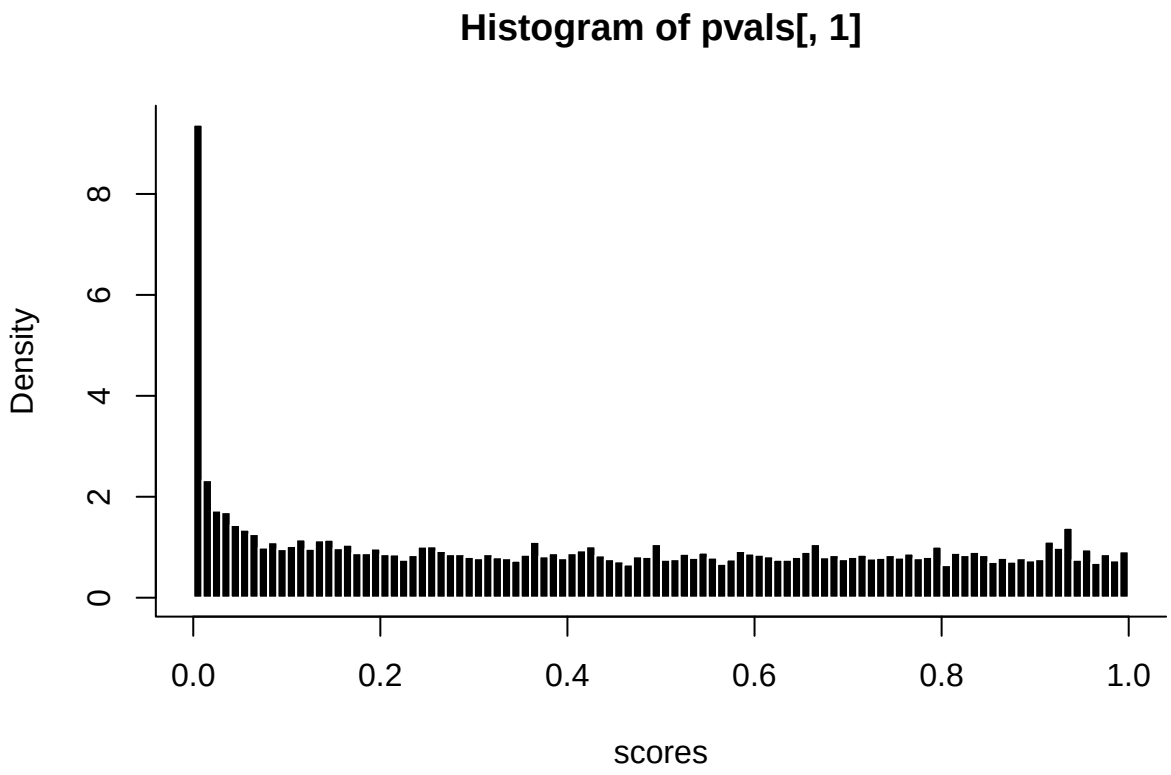
```
##     baseMean          log2FoldChange        lfcSE             stat
##  Min.   :     0.00   Min.   :-7.7173   Min.   :0.1631   Min.   :-27.1854
##  1st Qu.:    20.57   1st Qu.:-0.2990   1st Qu.:0.2291   1st Qu.: -0.8169
##  Median :   140.12   Median : 0.0166   Median :0.3059   Median :  0.0447
##  Mean   :   575.65   Mean   : 0.0249   Mean   :0.6431   Mean   : -0.1005
##  3rd Qu.:   446.26   3rd Qu.: 0.3713   3rd Qu.:0.6259   3rd Qu.:  0.8506
##  Max.   :100749.48   Max.   : 6.6833   Max.   :4.9920   Max.   : 11.7450
##                      NA's   :418       NA's   :418      NA's   :418
##      pvalue            padj
##  Min.   :0.0000   Min.   :0.0000
##  1st Qu.:0.1268   1st Qu.:0.3963
##  Median :0.4044   Median :0.7240
##  Mean   :0.4266   Mean   :0.6226
##  3rd Qu.:0.7035   3rd Qu.:0.8911
##  Max.   :0.9999   Max.   :0.9999
##  NA's   :418      NA's   :2600
```

```
head(count_results_svRNA184)
```

```
##       baseMean log2FoldChange     lfcSE      stat        pvalue          padj
## MX1   1535.804      -5.572993 0.2049993 -27.18543 9.658759e-163 1.344016e-158
## OASL  1095.279      -5.545498 0.2154467 -25.73953 4.221962e-146 2.937430e-142
## IFIT1 4235.123      -5.120843 0.2050420 -24.97460 1.154374e-137 5.354371e-134
## ISG15 5057.195      -4.547792 0.1859176 -24.46133 3.812414e-132 1.326243e-128
## IFIT3 1694.756      -5.012684 0.2065019 -24.27427 3.666006e-130 1.020250e-126
## HELZ2 2747.630      -3.838123 0.1768202 -21.70636 1.786523e-104 4.143245e-101
```

```
# HISTOGRAM
pvals = count_results_svRNA184["pvalue"]
hist(
    pvals[, 1],
    prob = TRUE,
```

```
    col = "black",
    border = "white",
    xlab = "scores",
    breaks = 100
)
box(bty = "l")
# Draw density function (assuming normal dist)
score_mean = mean(pvals[, 1])
score_sd   = sd(pvals[, 1])
curve(
    dnorm(x, mean = score_mean, sd = score_sd),
    add = TRUE,
    col = "red",
    lwd = 2
)
```




```
# Let's take a look to some genes
# count_mat_28v0["ENSG00000128567.16_PODXL",]
# count_mat_28v0["ENSG00000185559.14_DLK1",]
```

Apply DESeq2 to X3_9 (influenza A/Puerto Rico/8/1934)

```
tbl_merge = merge(X3_9_mock, X3_9_wt, by="row.names", all=TRUE)
rownames(tbl_merge) <- tbl_merge[,1] # First column contains the row names
tbl_merge = tbl_merge %>% select(-contains(c("Row.names"))) # DROP extra data
print(head(tbl_merge))
```

```
##          X3_9_mock1_13 X3_9_mock2_14 X3_9_wt1_15 X3_9_wt2_16
## A1BG                21            21           5           8
## A1BG-AS1             0             6           2           1
## A1CF                 0             0           0           0
## A2M                  0             1           0           1
## A2M-AS1             12            26           7           2
```

```
## A2ML1                0             0           0           0
```

```
# Conteos
aux_classes = rep(0, times = ncol(tbl_merge)) # CLASSIFY infected samples as 0s
aux_classes[grep(pattern = "mock", x = colnames(tbl_merge))] = 1 # and mocks as 1s
aux_classes
```

```
## [1] 1 1 0 0
```

```
colnames(tbl_merge)
```

```
## [1] "X3_9_mock1_13" "X3_9_mock2_14" "X3_9_wt1_15"   "X3_9_wt2_16"
```

```
count_results_X3_9 = DESeq_func(matrix_c = tbl_merge, classes_c = aux_classes)
```

```
## estimating size factors
```

```
## estimating dispersions
```

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

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

```
## final dispersion estimates
```

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

```
count_results_X3_9 = count_results_X3_9[order(count_results_X3_9$pvalue),]
summary(count_results_X3_9)
```

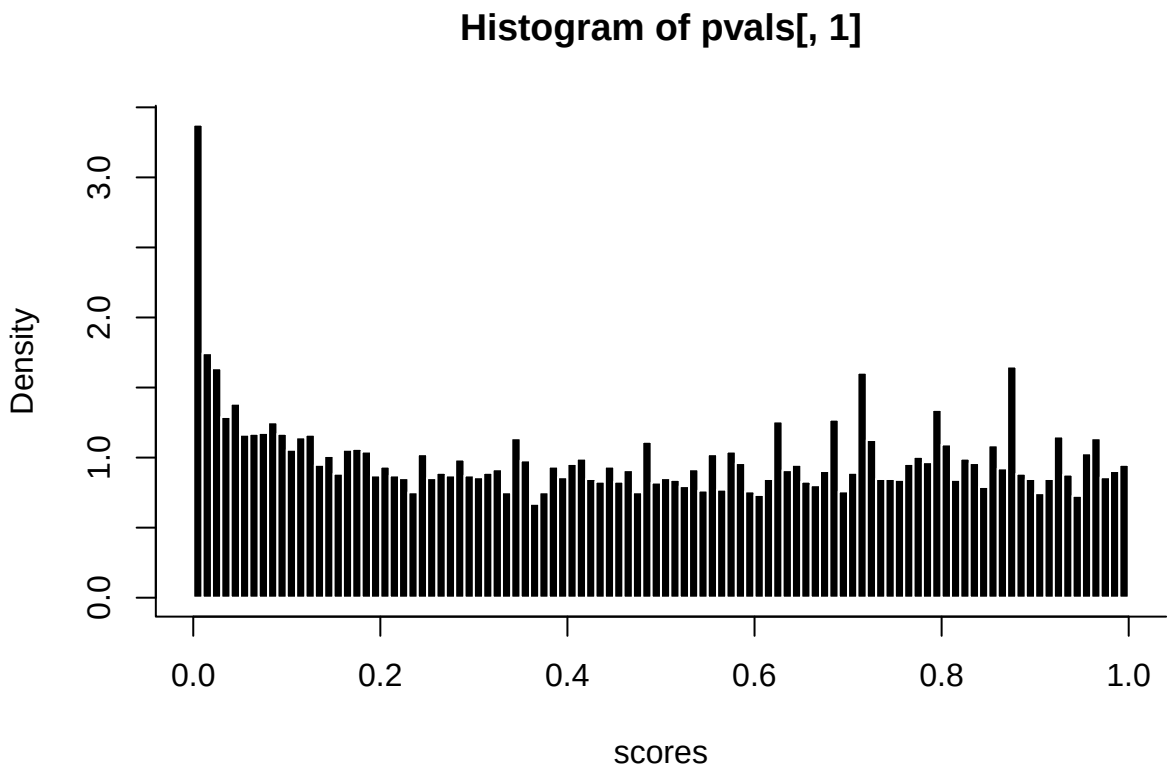
```
##     baseMean        log2FoldChange        lfcSE              stat
##  Min.   :    0.00   Min.   :-6.8682   Min.   :0.2386   Min.   :-5.0992
##  1st Qu.:    6.58   1st Qu.:-0.3830   1st Qu.:0.3352   1st Qu.:-0.7039
##  Median :   76.35   Median : 0.0014   Median :0.4516   Median : 0.0025
##  Mean   :  267.67   Mean   : 0.0576   Mean   :1.0106   Mean   : 0.0420
##  3rd Qu.:  242.78   3rd Qu.: 0.4995   3rd Qu.:1.0701   3rd Qu.: 0.7245
##  Max.   :35035.18   Max.   : 6.0135   Max.   :4.9730   Max.   : 5.8261
##                     NA's   :701       NA's   :701      NA's   :701
##      pvalue            padj
##  Min.   :0.0000   Min.   :0.000
##  1st Qu.:0.1929   1st Qu.:0.607
##  Median :0.4750   Median :0.809
##  Mean   :0.4749   Mean   :0.742
##  3rd Qu.:0.7431   3rd Qu.:0.929
##  Max.   :1.0000   Max.   :1.000
##  NA's   :701      NA's   :3760
```

```
head(count_results_X3_9)
```

```
##             baseMean log2FoldChange     lfcSE      stat       pvalue
## RPL39      488.84050       1.734375 0.2976903  5.826106 5.673559e-09
## MAFK       228.13370       2.126322 0.3824744  5.559385 2.707271e-08
## CPLX2     6171.97802       1.384211 0.2553515  5.420807 5.933062e-08
## NEAT1     1145.10887       1.363436 0.2604940  5.234037 1.658470e-07
## ZDHHC24     55.77728       3.486370 0.6770468  5.149378 2.613512e-07
## LINC00641   79.21125      -2.306046 0.4522335 -5.099237 3.410247e-07
##                   padj
## RPL39     7.236624e-05
## MAFK      1.726562e-04
## CPLX2     2.522540e-04
## NEAT1     5.288447e-04
## ZDHHC24   6.667068e-04
## LINC00641 7.249616e-04
```

```
# HISTOGRAM
pvals = count_results_X3_9["pvalue"]
hist(
    pvals[, 1],
    prob = TRUE,
```

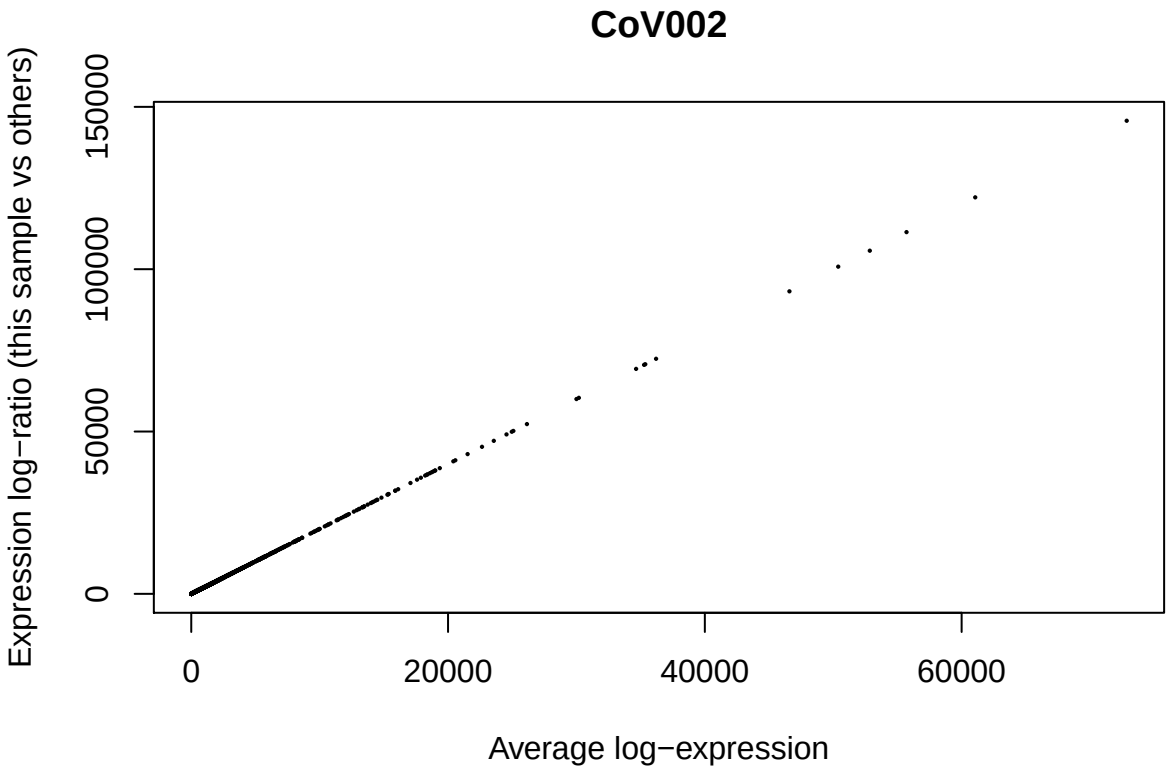
```
    col = "black",
    border = "white",
    xlab = "scores",
    breaks = 100
)
box(bty = "l")
# Draw density function (assuming normal dist)
score_mean = mean(pvals[, 1])
score_sd   = sd(pvals[, 1])
curve(
    dnorm(x, mean = score_mean, sd = score_sd),
    add = TRUE,
    col = "red",
    lwd = 2
)
```



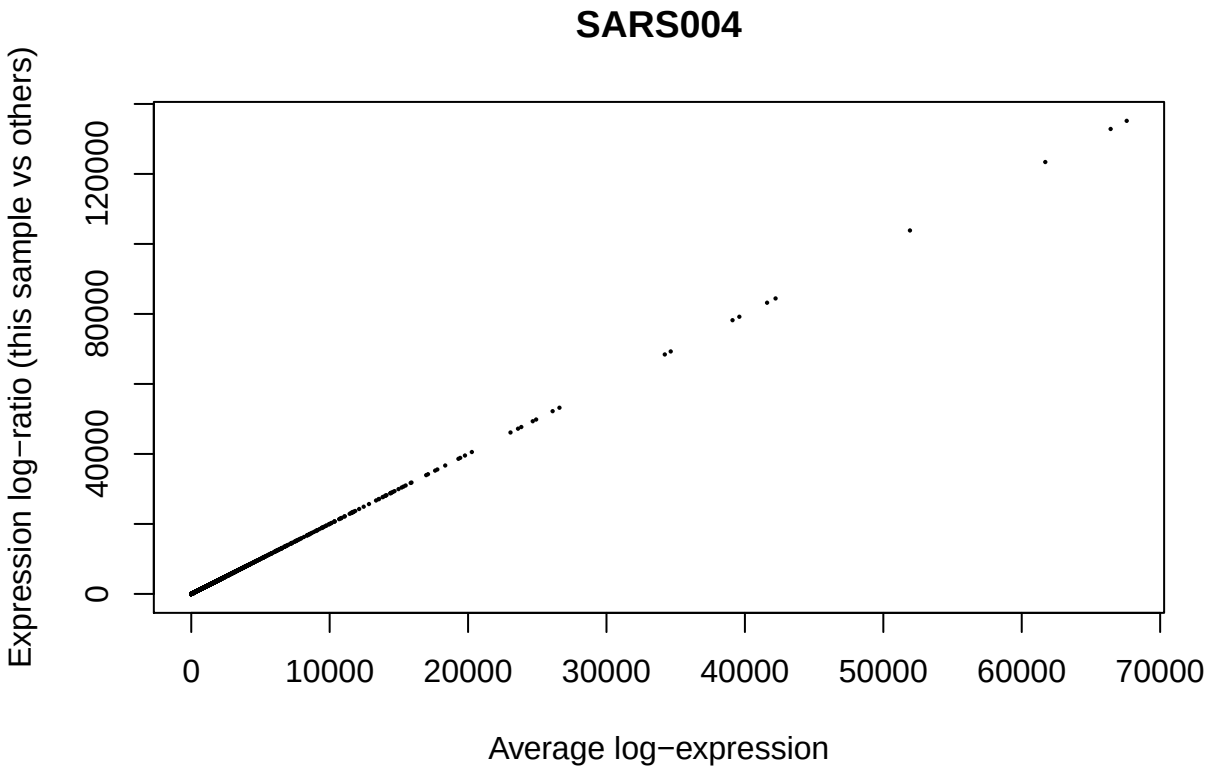

```
# Let's take a look to some genes
# count_mat_28v0["ENSG00000128567.16_PODXL",]
# count_mat_28v0["ENSG00000185559.14_DLK1",]
```

Comparar CoV002 (SARS-related coronavirus 2) vs SARS004 (SARS), svRNA184 (human respiratory syncytial virus) y X3_9 (influenza A/Puerto Rico/8/1934)

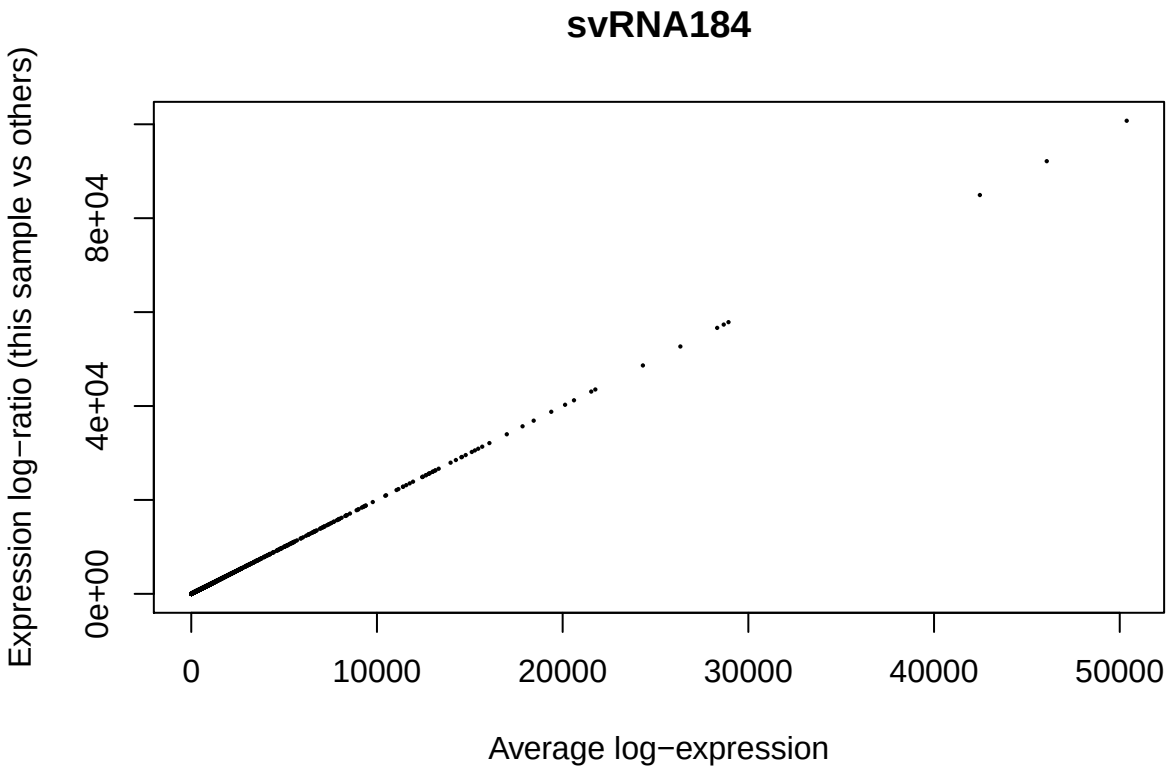
```
plotMA(count_results_CoV002, main = "CoV002")
```



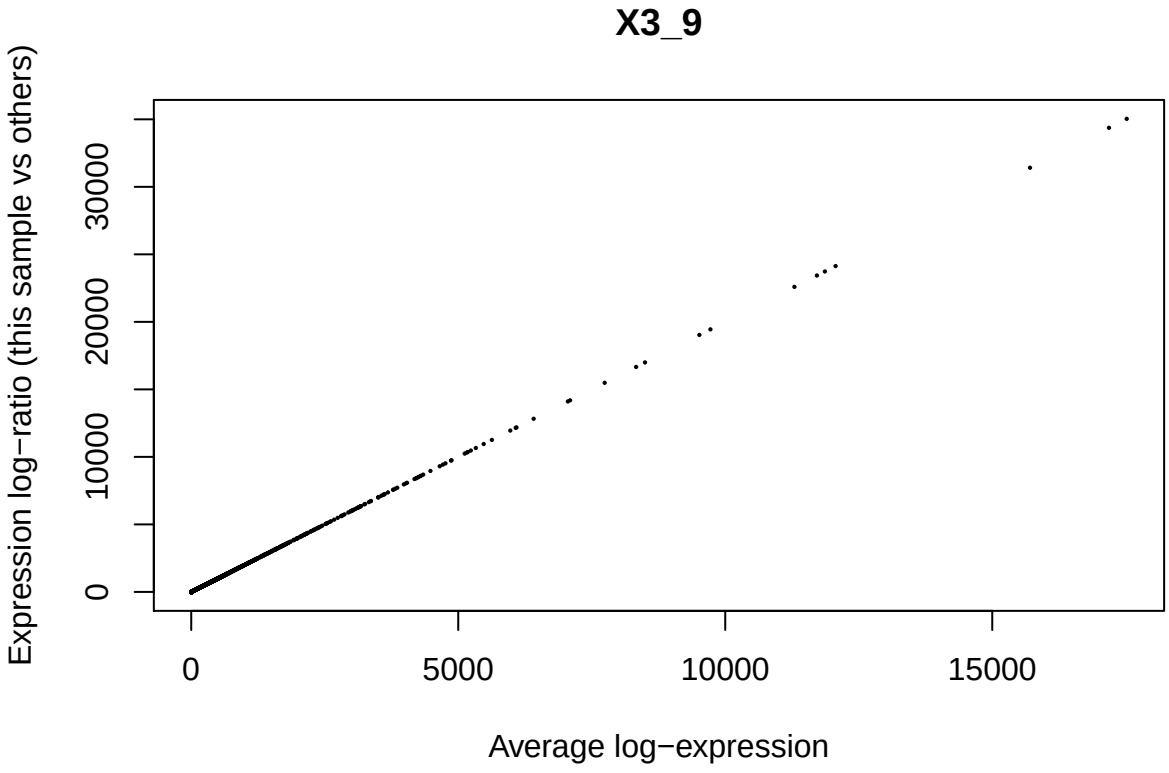

```
plotMA(count_results_SARS004, main = "SARS004")
```



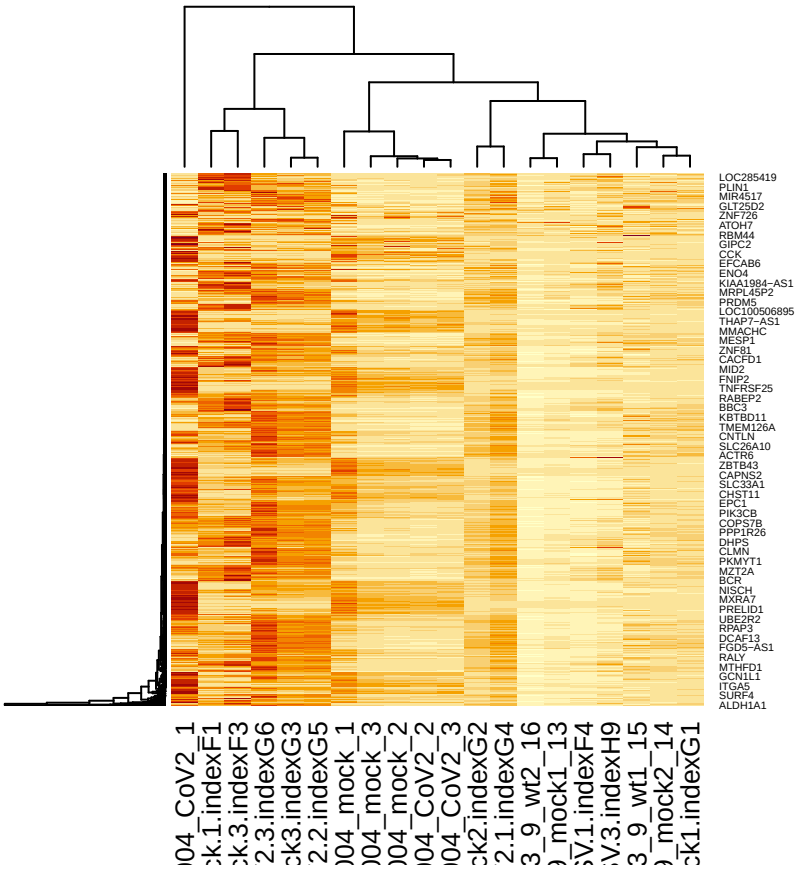

```
plotMA(count_results_svRNA184, main = "svRNA184")
```



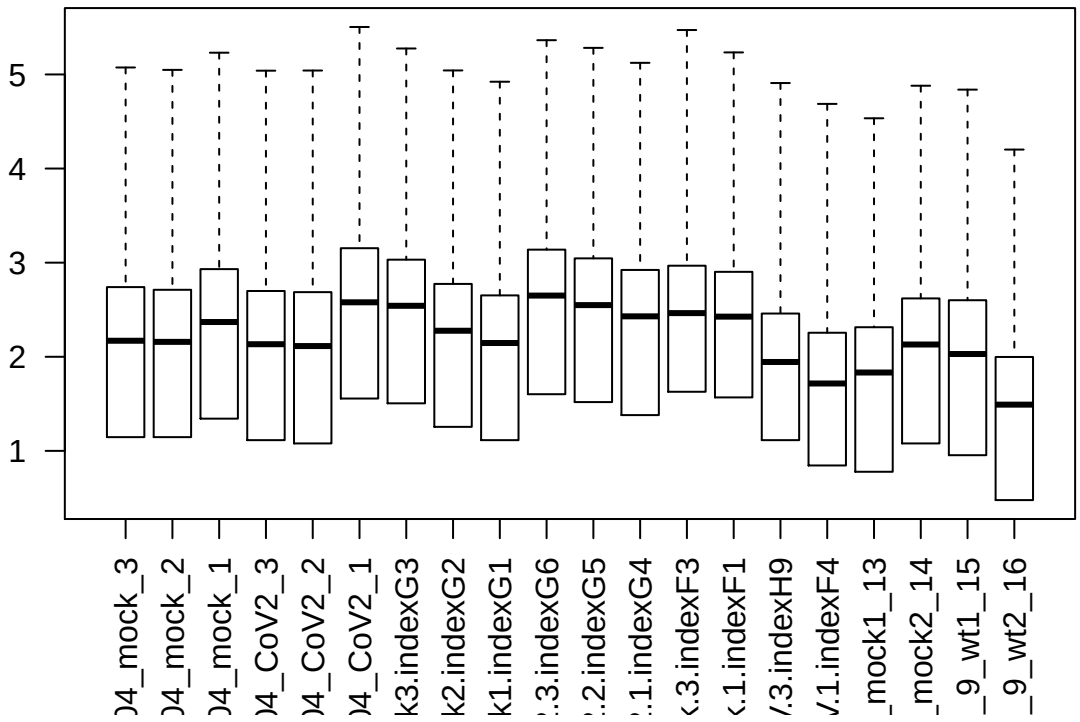

```
plotMA(count_results_X3_9, main = "X3_9")
```




```
heatmap(as.matrix(datos_covid))
```




```
boxplot(log10(datos_covid), range=0, las=2)
```



TNFSF15 y EDN1 son los genes que están presentes al comparar SARS-CoV-2 con otros virus. DESeq2 muestra que los datos proporcionados sugieren que las drogas que aumentan la respuesta antiviral pueden ser una opción efectiva en el tratamiento de COVID-19.