Assignment6b

Sameera Boppana

2/28/2022

```
library(DESeq2)
```

```
## Warning: package 'DESeq2' was built under R version 4.0.3
```

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

```
## Warning: package 'S4Vectors' was built under R version 4.0.3
```

```
## Loading required package: stats4
```

```
## Loading required package: BiocGenerics
```

```
## Warning: package 'BiocGenerics' was built under R version 4.0.5
```

```
## 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: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.max, which.min
```

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

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

```
## Loading required package: IRanges
```

```
## Warning: package 'IRanges' was built under R version 4.0.3
```

```
## Loading required package: GenomicRanges
```

```
## Warning: package 'GenomicRanges' was built under R version 4.0.3
```

```
## Loading required package: GenomeInfoDb
```

```
## Warning: package 'GenomeInfoDb' was built under R version 4.0.5
```

```
## Loading required package: SummarizedExperiment
```

```
## Warning: package 'SummarizedExperiment' was built under R version 4.0.3
```

```
## Loading required package: MatrixGenerics
```

```
## Warning: package 'MatrixGenerics' was built under R version 4.0.3
```

```
## 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
```

```
## Warning: package 'Biobase' was built under R version 4.0.3
```

```
## 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
```

```
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(viridis)
```

```
## Loading required package: viridisLite
```

```
uninfected1_orig <- read.table('output1/abundance.tsv', header = T)
uninfected2_orig <- read.table('output2/abundance.tsv', header = T)
infected_orig <- read.table('output3/abundance.tsv', header = T)

uninfected1 <- data.frame(uninfected1_orig[,1], uninfected1_orig[,4])
names(uninfected1)[1] <- "gene_id"
names(uninfected1)[2] <- "sample1a"


uninfected2 <- data.frame(uninfected2_orig[,1], uninfected2_orig[,4])
names(uninfected2)[1] <- "gene_id"
names(uninfected2)[2] <- "sample1b"


infected <- data.frame(infected_orig[,1], infected_orig[,4])
names(infected)[1] <- "gene_id"
names(infected)[2] <- "sample2"


intermediate <- merge(uninfected1, uninfected2, by = "gene_id")
counts <- merge(intermediate, infected, by = "gene_id")

row.names(counts) <- counts$gene_id

counts <- counts[,c(2:4)]
names(counts) <- c('uninfected1', 'uninfected2', 'infected')
counts <- round(counts)
head(counts)
```

```
##           uninfected1 uninfected2 infected
## NM_000014        1818        2948        0
## NM_000015           0           0        0
## NM_000016          93          79       14
## NM_000017          75          60        0
## NM_000018        2724        1212       86
## NM_000019         218         219       25
```

```
samples <- data.frame('group' = c('uninfected', 'uninfected', 'infected'),
                      row.names = c('uninfected1', 'uninfected2', 'intected'))
samples
```

```
##                  group
## uninfected1 uninfected
## uninfected2 uninfected
## intected      infected
```

```
dds <- DESeqDataSetFromMatrix(counts, samples, design = ~ group)
```

```
## converting counts to integer mode
```

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

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

```
## estimating size factors
```

```
## estimating dispersions
```

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

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

```
## final dispersion estimates
```

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

```
uninfected_vs_infected <- results(dds, contrast = c('group', 'uninfected', 'infected'))
head(uninfected_vs_infected)
```

```
## log2 fold change (MLE): group uninfected vs infected
## Wald test p-value: group uninfected vs infected
## DataFrame with 6 rows and 6 columns
##            baseMean log2FoldChange     lfcSE       stat      pvalue       padj
##           <numeric>      <numeric> <numeric>  <numeric>   <numeric>  <numeric>
## NM_000014  892.0523     11.0925467   2.97042  3.7343331 0.000188213 0.00934339
## NM_000015    0.0000             NA        NA         NA          NA         NA
## NM_000016   45.9634     -0.0316369   2.31209 -0.0136832 0.989082698 0.99862976
## NM_000017   23.7452      5.8610590   3.34106  1.7542529 0.079387189         NA
## NM_000018  752.9791      1.7835970   2.04705  0.8712998 0.383590457 0.68236753
## NM_000019  106.2278      0.4990673   2.15836  0.2312256 0.817139547 0.94674247
```

```
infected_vs_uninfected <- results(dds, contrast = c('group', 'infected', 'uninfected'))
head(infected_vs_uninfected)
```

```
## log2 fold change (MLE): group infected vs uninfected
## Wald test p-value: group infected vs uninfected
## DataFrame with 6 rows and 6 columns
##            baseMean log2FoldChange     lfcSE       stat      pvalue       padj
##           <numeric>      <numeric> <numeric>  <numeric>   <numeric>  <numeric>
## NM_000014  892.0523    -11.0925467   2.97042 -3.7343331 0.000188213 0.00934339
## NM_000015    0.0000             NA        NA         NA          NA         NA
## NM_000016   45.9634      0.0316369   2.31209  0.0136832 0.989082698 0.99862976
## NM_000017   23.7452     -5.8610590   3.34106 -1.7542529 0.079387189         NA
## NM_000018  752.9791     -1.7835970   2.04705 -0.8712998 0.383590457 0.68236753
## NM_000019  106.2278     -0.4990673   2.15836 -0.2312256 0.817139547 0.94674247
```

```
infected_over_uninfected <- subset(infected_vs_uninfected, log2FoldChange > 0)
head(infected_over_uninfected)
```

```
## log2 fold change (MLE): group infected vs uninfected
## Wald test p-value: group infected vs uninfected
## DataFrame with 6 rows and 6 columns
##            baseMean log2FoldChange     lfcSE      stat    pvalue      padj
##           <numeric>      <numeric> <numeric> <numeric> <numeric> <numeric>
## NM_000016   45.9634      0.0316369   2.31209 0.0136832  0.989083  0.998630
## NM_000024  120.2267      0.9921971   2.14783 0.4619531  0.644115  0.873060
## NM_000026   24.0270      1.5910387   2.54188 0.6259293  0.531361        NA
## NM_000033   17.8552      0.2521843   2.70311 0.0932940  0.925670        NA
## NM_000046   51.1374      0.9970240   2.28356 0.4366094  0.662395  0.881493
## NM_000049   35.7384      0.2491992   2.38843 0.1043359  0.916903        NA
```

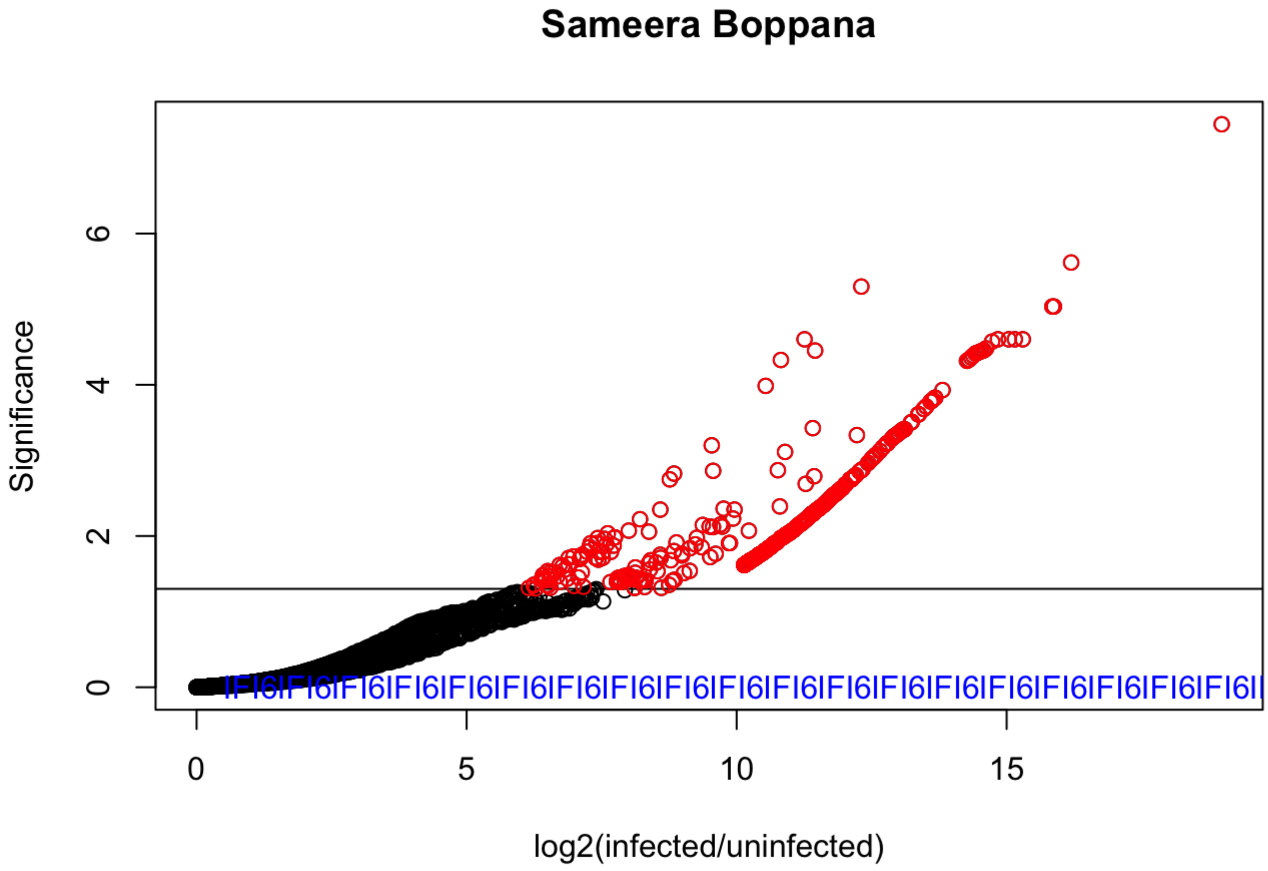
```
signif_diff_exp <- subset(infected_over_uninfected, padj < 0.05)
head(signif_diff_exp)
```

```
## log2 fold change (MLE): group infected vs uninfected
## Wald test p-value: group infected vs uninfected
## DataFrame with 6 rows and 6 columns
##                 baseMean log2FoldChange     lfcSE      stat      pvalue
##                <numeric>      <numeric> <numeric> <numeric>   <numeric>
## NM_000115        66.6204       11.00238   2.92365   3.76324 1.67727e-04
## NM_000206      7621.0570        6.81556   2.10506   3.23771 1.20495e-03
## NM_000314       179.8751       12.43533   2.79779   4.44470 8.80150e-06
## NM_000577     10481.1838        8.29707   2.78871   2.97524 2.92764e-03
## NM_000632        59.9584       10.85037   2.94529   3.68397 2.29629e-04
## NM_001001392    406.3845       13.61117   2.75527   4.94004 7.81052e-07
##                     padj
##                <numeric>
## NM_000115    0.00877446
## NM_000206    0.02789072
## NM_000314    0.00108082
## NM_000577    0.04730444
## NM_000632    0.01042357
## NM_001001392 0.00016271
```

```
subset(signif_diff_exp, row.names(signif_diff_exp) == 'NM_002038')
```

```
## log2 fold change (MLE): group infected vs uninfected
## Wald test p-value: group infected vs uninfected
## DataFrame with 1 row and 6 columns
##            baseMean log2FoldChange     lfcSE      stat      pvalue      padj
##           <numeric>      <numeric> <numeric> <numeric>   <numeric> <numeric>
## NM_002038   13814.3        7.29462     2.022   3.60762 0.000309017  0.012474
```

```
infected_over_uninfected <- data.frame(infected_over_uninfected)
signif_diff_exp <- data.frame(signif_diff_exp)
plot(x = infected_over_uninfected$log2FoldChange, y = -log10(infected_over_uninfected$pa
dj), xlab = "log2(infected/uninfected)", ylab = "Significance") + title("Sameera Boppan
a") + abline(h = -log10(0.05)) + points(x = signif_diff_exp$log2FoldChange, y = -log10(s
ignif_diff_exp$padj), col = "red") + text(x = row.names(signif_diff_exp) == 'NM_002038',
"IFI6", col = "blue")
```


Sameera Boppana
Significance
0
2
4
6
0
5
10
15
log2(infected/uninfected)
IFI6IFI6IFI6IFI6IFI6IFI6IFI6IFI6IFI6IFI6IFI6IFI6IFI6IFI6IFI6IFI6IFI6IFI6IFI6IFI6II

```
## integer(0)
```

```
uninfected1_tpm <- data.frame(uninfected1_orig[,1], uninfected1_orig[,5])
names(uninfected1_tpm)[1] <- "gene_id"
names(uninfected1_tpm)[2] <- "sample1a"


uninfected2_tpm <- data.frame(uninfected2_orig[,1], uninfected2_orig[,5])
names(uninfected2_tpm)[1] <- "gene_id"
names(uninfected2_tpm)[2] <- "sample1b"


infected_tpm <- data.frame(infected_orig[,1], infected_orig[,5])
names(infected_tpm)[1] <- "gene_id"
names(infected_tpm)[2] <- "sample2"


intermediate_tpm <- merge(uninfected1_tpm, uninfected2_tpm, by = "gene_id")
tpm <- merge(intermediate_tpm, infected_tpm, by = "gene_id")

row.names(tpm) <- tpm$gene_id

tpm <- tpm[,c(2:4)]
names(tpm) <- c('uninfected1', 'uninfected2', 'infected')
tpm <- round(tpm)
head(tpm)
```

```
##           uninfected1 uninfected2 infected
## NM_000014          52         184        0
## NM_000015           0           0        0
## NM_000016           6          11        4
## NM_000017           6          10        0
## NM_000018         173         168       26
## NM_000019          21          45       11
```

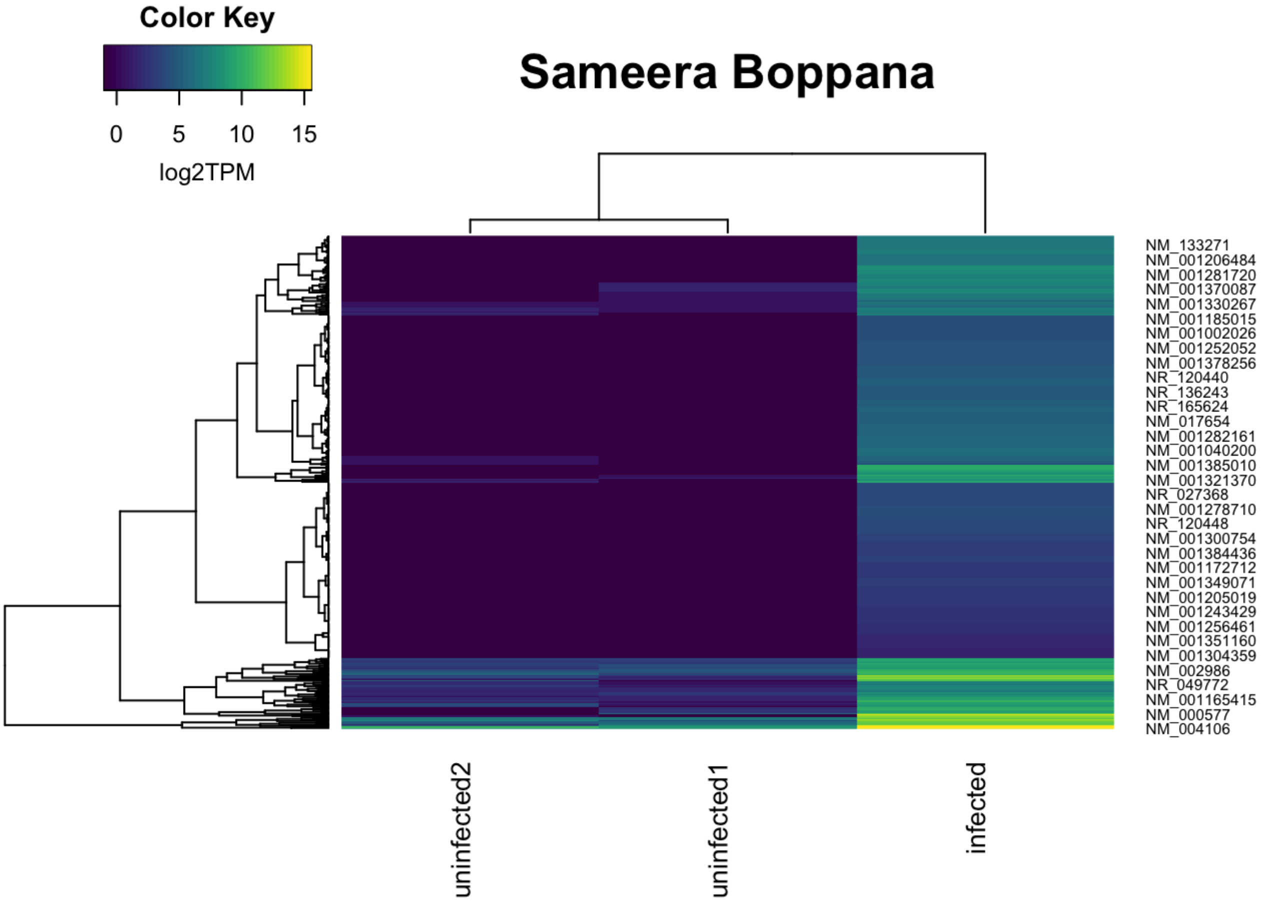
```
tpm_diff_exp <- subset(tpm, row.names(tpm) %in% row.names(signif_diff_exp))
head(tpm_diff_exp)
```

```
##              uninfected1 uninfected2 infected
## NM_000115              0           0        9
## NM_000206             73          25     3144
## NM_000314              0           0       12
## NM_000577             39           0     3805
## NM_000632              0           0        7
## NM_001001392           0           0       63
```

```
tpm_diff_exp$uninfected1 <- log2(tpm_diff_exp$uninfected1 + 0.5)
tpm_diff_exp$uninfected2 <- log2(tpm_diff_exp$uninfected2 + 0.5)
tpm_diff_exp$infected <- log2(tpm_diff_exp$infected + 0.5)
head(tpm_diff_exp)
```

```
##              uninfected1 uninfected2  infected
## NM_000115      -1.000000   -1.000000  3.247928
## NM_000206       6.199672    4.672425 11.618615
## NM_000314      -1.000000   -1.000000  3.643856
## NM_000577       5.303781   -1.000000 11.893870
## NM_000632      -1.000000   -1.000000  2.906891
## NM_001001392   -1.000000   -1.000000  5.988685
```

```
par(oma=c(1,1,1,1))
heatmap.2(as.matrix(tpm_diff_exp), dendrogram = "both", cexCol = 1, symkey = FALSE, symb
reaks = FALSE, main = "Sameera Boppana", key = TRUE, density.info = 'none', key.xlab= "l
og2TPM", trace = "none",col=viridis(50))
```


Color Key
log2TPM
Sameera Boppana
NM_133271
NM_001206484
NM_001281720
NM_001370087
NM_001330267
NM_001185015
NM_001002026
NM_001252052
NM_001378256
NR_120440
NR_136243
NR_165624
NM_017654
NM_001282161
NM_001040200
NM_001385010
NM_001321370
NR_027368
NM_001278710
NR_120448
NM_001300754
NM_001384436
NM_001172712
NM_001349071
NM_001205019
NM_001243429
NM_001256461
NM_001351160
NM_001304359
NM_002986
NR_049772
NM_001165415
NM_000577
NM_004106
uninfected2
uninfected1
infected