Class13:Transcriptomics and the analysis of RNA-Seq data

Jihyun In(A16955363)

Background

Today we will analyze some RNA Sequencing data on the effects ofa common steroid drug on airway cell lines.

There are two main inputs we need for this analysis:

- `countData`: counts for genes in rows with experiments in the columns
- `colData`: metadata that tells us about the desgn of the experiment (e.e.what is in the columns of `countData`)

Import

```
# Complete the missing code
counts <- read.csv("airway_scaledcounts.csv", row.names=1)
metadata <-  read.csv("airway_metadata.csv")
```

Q1. How many genes are in this dataset?

```
head(counts)
```

```
                SRR1039508 SRR1039509 SRR1039512 SRR1039513 SRR1039516
ENSG00000000003        723        486        904        445       1170
ENSG00000000005          0          0          0          0          0
ENSG00000000419        467        523        616        371        582
ENSG00000000457        347        258        364        237        318
ENSG00000000460         96         81         73         66        118
ENSG00000000938          0          0          1          0          2
                SRR1039517 SRR1039520 SRR1039521
```

```
ENSG00000000003       1097        806        604
ENSG00000000005          0          0          0
ENSG00000000419        781        417        509
ENSG00000000457        447        330        324
ENSG00000000460         94        102         74
ENSG00000000938          0          0          0
```

```
nrow(counts)
```

```
[1] 38694
```

38694 genes are in this dataset.

Q2. How many 'control' cell lines do we have?

```
#sum(metadata$dex == "control")
table(metadata$dex)
```

```
control treated
      4       4
```

We have 4 control cell lines.

Toy differential gene expression

Let's try finding the average or mean of the "control" and "treated" cp;umns and see if they differ.

- First we need to find all "control columns
- extract just the "control" values for each gene
- caclulate the `mean()` for each gene"control" values

```
control <- metadata[metadata[,"dex"]=="control",]
control.counts <- counts[ ,control$id]
#control.counts
control.mean <- rowMeans(control.counts)
head(control.mean)
```

```
ENSG00000000003 ENSG00000000005 ENSG00000000419 ENSG00000000457 ENSG00000000460
         900.75             0.00           520.50           339.75            97.25
ENSG00000000938
           0.75
```

Q. Do the same for "treated" to get a `treated.mean()`

```
treated <- metadata[metadata[,"dex"]=="treated",]
treated.counts <- counts[ ,treated$id]
#treated.counts
treated.mean <- rowMeans(treated.counts)
head(treated.mean)
```

```
ENSG00000000003 ENSG00000000005 ENSG00000000419 ENSG00000000457 ENSG00000000460
         658.00             0.00           546.00           316.50            78.75
ENSG00000000938
           0.00
```

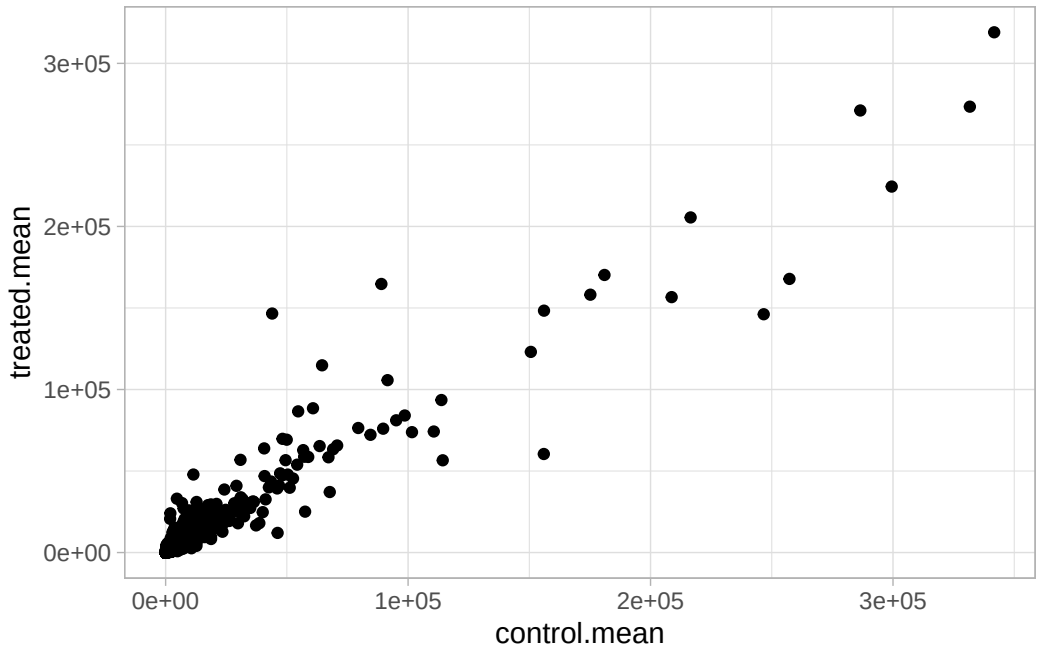
```
meancounts <- data.frame(control.mean, treated.mean)
head(meancounts)
```

```
                control.mean treated.mean
ENSG00000000003       900.75       658.00
ENSG00000000005         0.00         0.00
ENSG00000000419       520.50       546.00
ENSG00000000457       339.75       316.50
ENSG00000000460        97.25        78.75
ENSG00000000938         0.75         0.00
```

Q4. Make a plot of `control. mean` vs `treated.mean`

```
library(ggplot2)
```

```
ggplot(meancounts) + aes(control.mean, treated.mean) + geom_point() + theme_light()
```

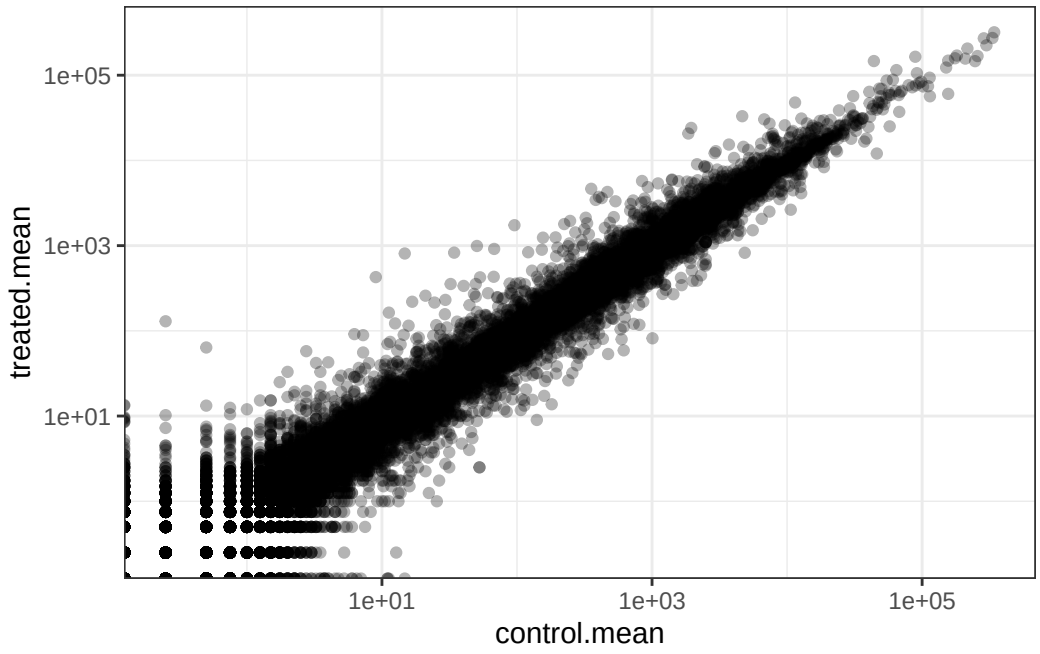


Now to plot this on a log-log scale:

```
ggplot(meancounts) + aes(control.mean, treated.mean) + geom_point(alpha=0.3) + theme_bw() + s
```

```
Warning in scale_x_log10(): log-10 transformation introduced infinite values.
```

```
Warning in scale_y_log10(): log-10 transformation introduced infinite values.
```

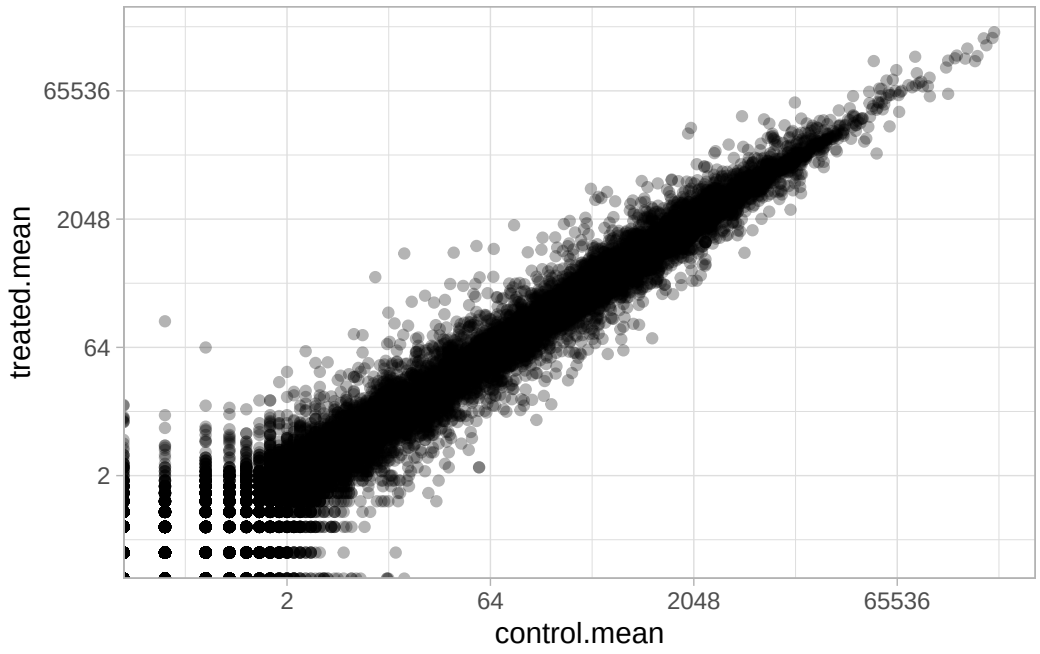


On a log2 scale instead:

```
ggplot(meancounts) + aes(control.mean, treated.mean) + geom_point(alpha = 0.3) + theme_light
```

```
Warning in scale_x_continuous(trans = "log2"): log-2 transformation introduced
infinite values.
```

```
Warning in scale_y_continuous(trans = "log2"): log-2 transformation introduced
infinite values.
```



Why use the log_2 scale?

```
#treated/control

log2(40/20) #Doubling the amount
```

```
[1] 1
```

```
log2(10/20) #halving the amount
```

```
[1] -1
```

So I'd assume we want to be more sensitive to changes than 10-fold changes??

A commmon "rule-of-thumb" is to focus o genes with a log2 "fold-change" of +2 as so-called **up regulated** and -2 as **down regulated**

Let's add a log2 fold-change value to our `meancounts` data frame.

```
meancounts$log2fc <- log2(meancounts$treated.mean/meancounts$control.mean)
head(meancounts)
```

```
                control.mean treated.mean      log2fc
ENSG00000000003       900.75       658.00 -0.45303916
ENSG00000000005         0.00         0.00         NaN
ENSG00000000419       520.50       546.00  0.06900279
ENSG00000000457       339.75       316.50 -0.10226805
ENSG00000000460        97.25        78.75 -0.30441833
ENSG00000000938         0.75         0.00        -Inf
```

Q. Remove any "zero count" genes from our dataset for further analysis

```
to.keep <- rowSums(meancounts[,1:2]==0) == 0
sum(to.keep)
```

```
[1] 21817
```

```
mycounts <- meancounts[to.keep,]
head(mycounts)
```

```
                control.mean treated.mean      log2fc
ENSG00000000003       900.75       658.00 -0.45303916
ENSG00000000419       520.50       546.00  0.06900279
ENSG00000000457       339.75       316.50 -0.10226805
ENSG00000000460        97.25        78.75 -0.30441833
ENSG00000000971      5219.00      6687.50  0.35769358
ENSG00000001036      2327.00      1785.75 -0.38194109
```

Q. How many genes are "up" regulated at a log2fc threshold of +2?

```
sum(mycounts$log2fc >= 2)
```

```
[1] 314
```

Q. How many genes are "down" regulated at a log2fc threshold of -2?

```
sum(mycounts$log2fc <= (-2))
```

```
[1] 485
```

We need more stats to see how trustworthy our results are.

Setting up for DESeq

LEt's do this properly and consider the stats - are the differences in the means significant?

```
library(DESeq2)
citation("DESeq2")
```

```
To cite package 'DESeq2' in publications use:

  Love, M.I., Huber, W., Anders, S. Moderated estimation of fold change
  and dispersion for RNA-seq data with DESeq2 Genome Biology 15(12):550
  (2014)

A BibTeX entry for LaTeX users is

  @Article{,
    title = {Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2
    author = {Michael I. Love and Wolfgang Huber and Simon Anders},
    year = {2014},
    journal = {Genome Biology},
    doi = {10.1186/s13059-014-0550-8},
    volume = {15},
    issue = {12},
    pages = {550},
  }
```

the first function we will use from this package sets up the input in the particular format that DESeq wants:

```
dds <- DESeqDataSetFromMatrix(countData=counts, colData = metadata, design=~dex)
```

```
converting counts to integer mode

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

```
dds
```

```
class: DESeqDataSet
dim: 38694 8
metadata(1): version
assays(1): counts
rownames(38694): ENSG00000000003 ENSG00000000005 ... ENSG00000283120
  ENSG00000283123
rowData names(0):
colnames(8): SRR1039508 SRR1039509 ... SRR1039520 SRR1039521
colData names(4): id dex celltype geo_id
```

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

```
estimating size factors
```

```
estimating dispersions
```

```
gene-wise dispersion estimates
```

```
mean-dispersion relationship
```

```
final dispersion estimates
```

```
fitting model and testing
```

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

```
head(res)
```

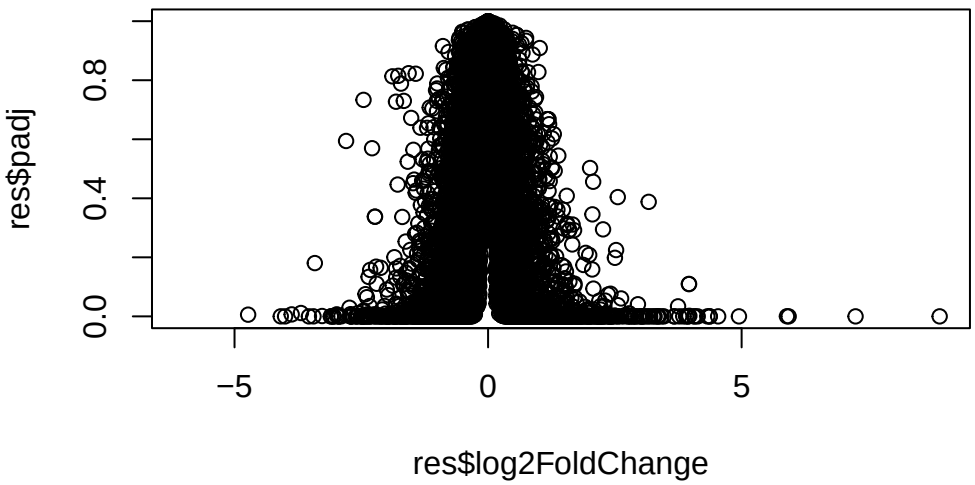
```
log2 fold change (MLE): dex treated vs control
Wald test p-value: dex treated vs control
DataFrame with 6 rows and 6 columns
                  baseMean log2FoldChange     lfcSE      stat    pvalue
                 <numeric>      <numeric> <numeric> <numeric> <numeric>
ENSG00000000003 747.194195     -0.3507030  0.168246 -2.084470 0.0371175
ENSG00000000005   0.000000             NA        NA        NA        NA
ENSG00000000419 520.134160      0.2061078  0.101059  2.039475 0.0414026
ENSG00000000457 322.664844      0.0245269  0.145145  0.168982 0.8658106
ENSG00000000460  87.682625     -0.1471420  0.257007 -0.572521 0.5669691
ENSG00000000938   0.319167     -1.7322890  3.493601 -0.495846 0.6200029
```

```
                     padj
                  <numeric>
ENSG00000000003  0.163035
ENSG00000000005        NA
ENSG00000000419  0.176032
ENSG00000000457  0.961694
ENSG00000000460  0.815849
ENSG00000000938        NA
```

Result figure: volcano plots

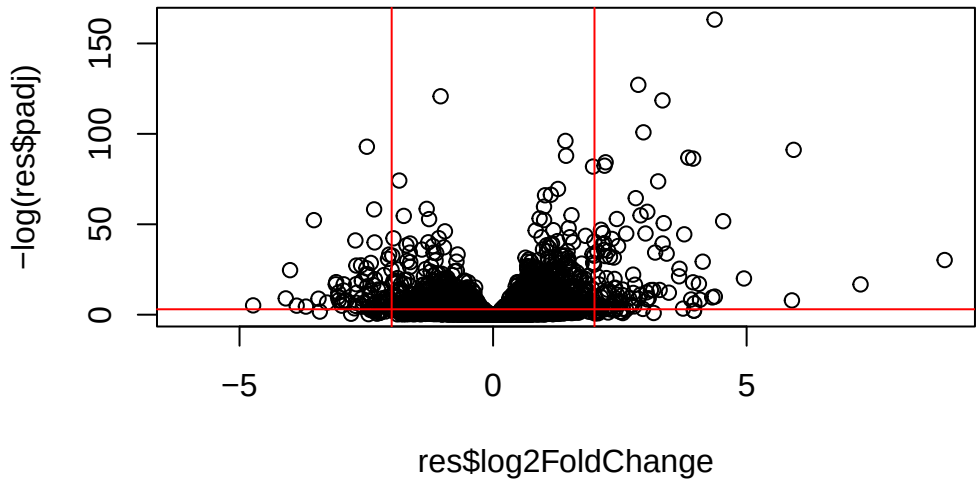
Plot of the log2FC vs P-value

```
plot(res$log2FoldChange, res$padj)
```

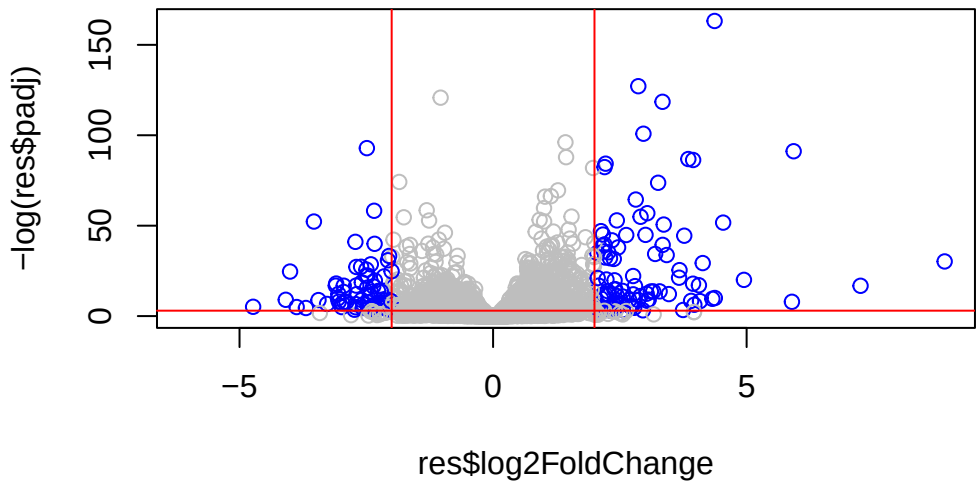


This p-value data is heavily skewed, so let's log trasform it.

```
plot(res$log2FoldChange, -log(res$padj)) #flipped y axis
abline(v=-2, col = "red")
abline(v=+2, col = "red")
abline(h=-log(0.05), col="red")
```



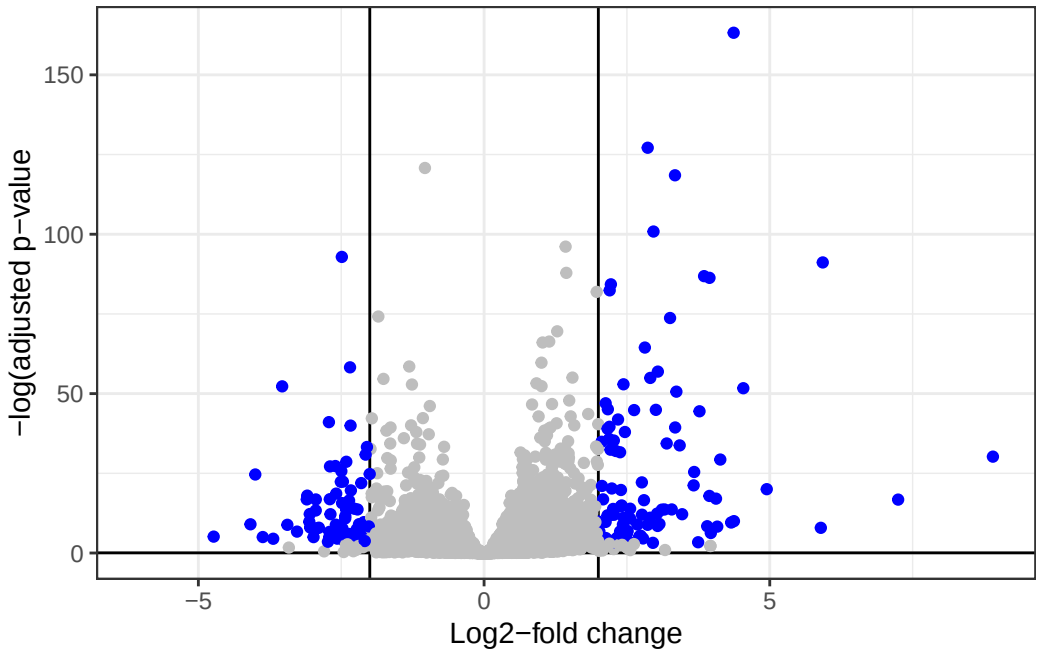

```
mycols <- rep("gray", nrow(res))
mycols[res$log2FoldChange <= -2] <- "blue"
mycols[res$log2FoldChange >= +2] <- "blue"

mycols[res$padj >= 0.05] <- "gray"

plot(res$log2FoldChange, -log(res$padj), col = mycols) #flipped y axis
abline(v=-2, col = "red")
abline(v=+2, col = "red")
abline(h=-log(0.05), col="red")
```



Q. Make a ggplot volcano plot

```
ggplot(res) +
  aes(log2FoldChange, -log(padj)) +
  geom_vline(xintercept = -2) +
  geom_vline(xintercept = 2) +
  geom_hline(yintercept = 0.05) +
  geom_point( col = mycols) +
  theme_bw() +
  labs(x = "Log2-fold change", y="-log(adjusted p-value")
```

```
Warning: Removed 23549 rows containing missing values or values outside the scale range
(`geom_point()`).
```



Gene notation

We first need to add gene symbols (eg. HBB etc.) so we know what we're working with. We need to "translate" between ENSEMBLE ids that we have in the names of "ids"

```
head(rownames(res))
```

```
[1] "ENSG00000000003" "ENSG00000000005" "ENSG00000000419" "ENSG00000000457"
[5] "ENSG00000000460" "ENSG00000000938"
```

Install from bioconductor with `BiocManager::install("AnnotationDbi")`

```
library(AnnotationDbi)
library(org.Hs.eg.db)
```

What differnet database ID types can I translate between?

```
columns(org.Hs.eg.db)
```

```
 [1] "ACCNUM"       "ALIAS"        "ENSEMBL"      "ENSEMBLPROT"  "ENSEMBLTRANS"
 [6] "ENTREZID"     "ENZYME"       "EVIDENCE"     "EVIDENCEALL"  "GENENAME"
[11] "GENETYPE"     "GO"           "GOALL"        "IPI"          "MAP"
[16] "OMIM"         "ONTOLOGY"     "ONTOLOGYALL"  "PATH"         "PFAM"
[21] "PMID"         "PROSITE"      "REFSEQ"       "SYMBOL"       "UCSCKG"
[26] "UNIPROT"
```

Let's "map" between "ENSEMBL" and "SYMBOL" (i.e. gene symbol)

```
res$symbol <- mapIds(x=org.Hs.eg.db,keys=rownames(res),
       keytype="ENSEMBL",
       column="SYMBOL")
```

```
'select()' returned 1:many mapping between keys and columns
```

```
head(res)
```

```
log2 fold change (MLE): dex treated vs control
Wald test p-value: dex treated vs control
DataFrame with 6 rows and 7 columns
                  baseMean log2FoldChange     lfcSE      stat    pvalue
                 <numeric>      <numeric> <numeric> <numeric> <numeric>
ENSG00000000003 747.194195     -0.3507030  0.168246 -2.084470 0.0371175
ENSG00000000005   0.000000             NA        NA        NA        NA
ENSG00000000419 520.134160      0.2061078  0.101059  2.039475 0.0414026
ENSG00000000457 322.664844      0.0245269  0.145145  0.168982 0.8658106
ENSG00000000460  87.682625     -0.1471420  0.257007 -0.572521 0.5669691
ENSG00000000938   0.319167     -1.7322890  3.493601 -0.495846 0.6200029
```

```
                    padj      symbol
                <numeric> <character>
ENSG00000000003  0.163035      TSPAN6
ENSG00000000005        NA        TNMD
ENSG00000000419  0.176032        DPM1
ENSG00000000457  0.961694       SCYL3
ENSG00000000460  0.815849       FIRRM
ENSG00000000938        NA         FGR
```

add a few more ID mappings including "GENENAME" and "ENTREZID".

```
res$name <- mapIds(x=org.Hs.eg.db,keys=rownames(res),
       keytype="ENSEMBL",
       column="GENENAME")
```

```
'select()' returned 1:many mapping between keys and columns
```

```
res$entrez <- mapIds(x=org.Hs.eg.db,keys=rownames(res),
       keytype="ENSEMBL",
       column="ENTREZID")
```

```
'select()' returned 1:many mapping between keys and columns
```

```
head(res)
```

```
log2 fold change (MLE): dex treated vs control
Wald test p-value: dex treated vs control
DataFrame with 6 rows and 9 columns
                  baseMean log2FoldChange     lfcSE      stat    pvalue
                 <numeric>      <numeric> <numeric> <numeric> <numeric>
ENSG00000000003 747.194195     -0.3507030  0.168246 -2.084470 0.0371175
ENSG00000000005   0.000000             NA        NA        NA        NA
ENSG00000000419 520.134160      0.2061078  0.101059  2.039475 0.0414026
ENSG00000000457 322.664844      0.0245269  0.145145  0.168982 0.8658106
ENSG00000000460  87.682625     -0.1471420  0.257007 -0.572521 0.5669691
ENSG00000000938   0.319167     -1.7322890  3.493601 -0.495846 0.6200029
                    padj      symbol                  name      entrez
                <numeric> <character>           <character> <character>
ENSG00000000003  0.163035      TSPAN6          tetraspanin 6        7105
ENSG00000000005        NA        TNMD            tenomodulin       64102
```

```
ENSG00000000419  0.176032        DPM1 dolichyl-phosphate m..        8813
ENSG00000000457  0.961694       SCYL3 SCY1 like pseudokina..       57147
ENSG00000000460  0.815849       FIRRM FIGNL1 interacting r..       55732
ENSG00000000938        NA         FGR FGR proto-oncogene, ..        2268
```

Be sure to save our annotated results to a file.

```
write.csv(res, file="my_annotated_results.csv")
```

Pathway Analysis

Find what biological pahtways my differentially expressed gene

Install the packages we need for pathway analysis: Run in your R console (i.e. not your Quarto doc!)

```
BiocManager::install( c("pathview", "gage", "gageData") )
```

```
library(gage)
```

```
library(gageData)
library(pathview)
```

```
################################################################################
Pathview is an open source software package distributed under GNU General
Public License version 3 (GPLv3). Details of GPLv3 is available at
http://www.gnu.org/licenses/gpl-3.0.html. Particullary, users are required to
formally cite the original Pathview paper (not just mention it) in publications
or products. For details, do citation("pathview") within R.

The pathview downloads and uses KEGG data. Non-academic uses may require a KEGG
license agreement (details at http://www.kegg.jp/kegg/legal.html).
################################################################################
```

```
data(kegg.sets.hs)

# Examine the first 2 pathways in this kegg set for humans
head(kegg.sets.hs, 2)
```

```
$`hsa00232 Caffeine metabolism`
[1] "10"   "1544" "1548" "1549" "1553" "7498" "9"

$`hsa00983 Drug metabolism - other enzymes`
 [1] "10"     "1066"   "10720"  "10941"  "151531" "1548"   "1549"   "1551"
 [9] "1553"   "1576"   "1577"   "1806"   "1807"   "1890"   "221223" "2990"
[17] "3251"   "3614"   "3615"   "3704"   "51733"  "54490"  "54575"  "54576"
[25] "54577"  "54578"  "54579"  "54600"  "54657"  "54658"  "54659"  "54963"
[33] "574537" "64816"  "7083"   "7084"   "7172"   "7363"   "7364"   "7365"
[41] "7366"   "7367"   "7371"   "7372"   "7378"   "7498"   "79799"  "83549"
[49] "8824"   "8833"   "9"      "978"
```

To run pathway analysiss we will use the `gage()` function and it requires a wee "vector of imprtance."We will use our Log2FC results from our `res` object:

```
foldchanges = res$log2FoldChange
names(foldchanges) = res$entrez
head(foldchanges)
```

```
       7105       64102        8813       57147       55732        2268
-0.35070302          NA  0.20610777  0.02452695 -0.14714205 -1.73228897
```

```
# Get the results
keggres = gage(foldchanges, gsets=kegg.sets.hs)
```

What is the returned `keggres` object

```
attributes(keggres)
```

```
$names
[1] "greater" "less"    "stats"
```

```
head(keggres$less)
```

```
                                                      p.geomean stat.mean
hsa05332 Graft-versus-host disease                 0.0004250461 -3.473346
hsa04940 Type I diabetes mellitus                  0.0017820293 -3.002352
hsa05310 Asthma                                    0.0020045888 -3.009050
hsa04672 Intestinal immune network for IgA production 0.0060434515 -2.560547
```

```
hsa05330 Allograft rejection                                0.0073678825 -2.501419
hsa04340 Hedgehog signaling pathway                         0.0133239547 -2.248547
                                                                   p.val      q.val
hsa05332 Graft-versus-host disease                          0.0004250461 0.09053483
hsa04940 Type I diabetes mellitus                           0.0017820293 0.14232581
hsa05310 Asthma                                             0.0020045888 0.14232581
hsa04672 Intestinal immune network for IgA production       0.0060434515 0.31387180
hsa05330 Allograft rejection                                0.0073678825 0.31387180
hsa04340 Hedgehog signaling pathway                         0.0133239547 0.47300039
                                                           set.size         exp1
hsa05332 Graft-versus-host disease                               40 0.0004250461
hsa04940 Type I diabetes mellitus                                42 0.0017820293
hsa05310 Asthma                                                  29 0.0020045888
hsa04672 Intestinal immune network for IgA production            47 0.0060434515
hsa05330 Allograft rejection                                     36 0.0073678825
hsa04340 Hedgehog signaling pathway                              56 0.0133239547
```

wE CAN PASS OUR FOLDCHANGES VECTOR(OUR RESULTS) together with any of these highlighted pathway IDs to see .

```
pathview(gene.data=foldchanges, pathway.id="hsa05310")
```

```
'select()' returned 1:1 mapping between keys and columns

Info: Working in directory C:/Users/User/Desktop/SCHOOL/BIMM143/class13_Transcriptomics

Info: Writing image file hsa05310.pathview.png
```

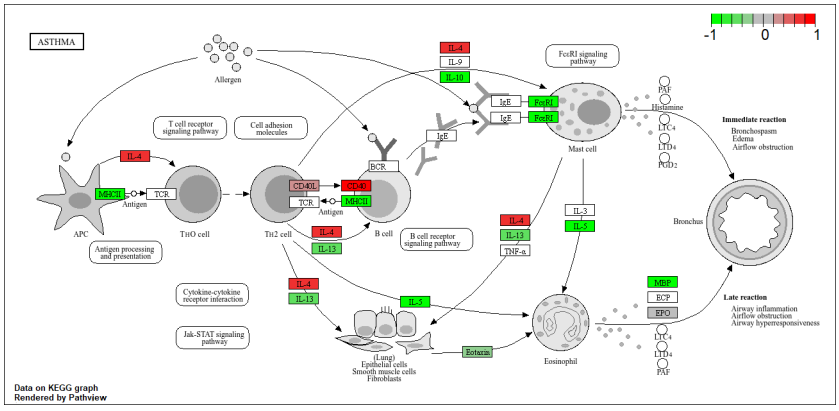


Figure 1: The asthma pathway overlaps with our differentially expressed genes.