Class 12- Transcriptomics and the analysis of RNA-Seq data

rachel galleta (A16859649)

Table of contents

Background

Today we will analyze some RNASeq data from Himes et al. on the effects of a common steroid (dexamethasone,) on airway smooth muscle cells (ASm cells) Are staring point is the “counts” data and “metadata” that contain the count values for each gene in their different experiments (i.e cell lines with or without drugs)

Data import

```r
counts <- read.csv("airway_scaledcounts.csv", row.names=1)
metadata <-read.csv("airway_metadata.csv")
```

```r
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
```

Q1. How many genes are in this dataset?

```r
nrow(counts)
```

```
[1] 38694
```

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

```r
sum(metadata$dex=="control")
```

```
[1] 4
```

Toy differential gene expression

To start our analysis lets calculate the mean for all genes in the "control" experiments.

1. extract all "control" columns form the counts objets
2. Calculate the mean for all rows (ie, genes) of these "control" columns 3-4. Do the same for "treated"
3. Compare these "control.mean" and "treated.mean" values.

```r
#1.
control.inds <- metadata$dex=="control"
control.counts<- counts[,control.inds]
```

```
#2.
control.means <- rowMeans( control.counts)
```

```
dim(control.counts)
```

```
[1] 38694     4
```

```
#3-4.
treated.inds <- metadata$dex=="treated"
treated.counts<- counts[,treated.inds]
treated.means <- rowMeans(treated.counts)
```

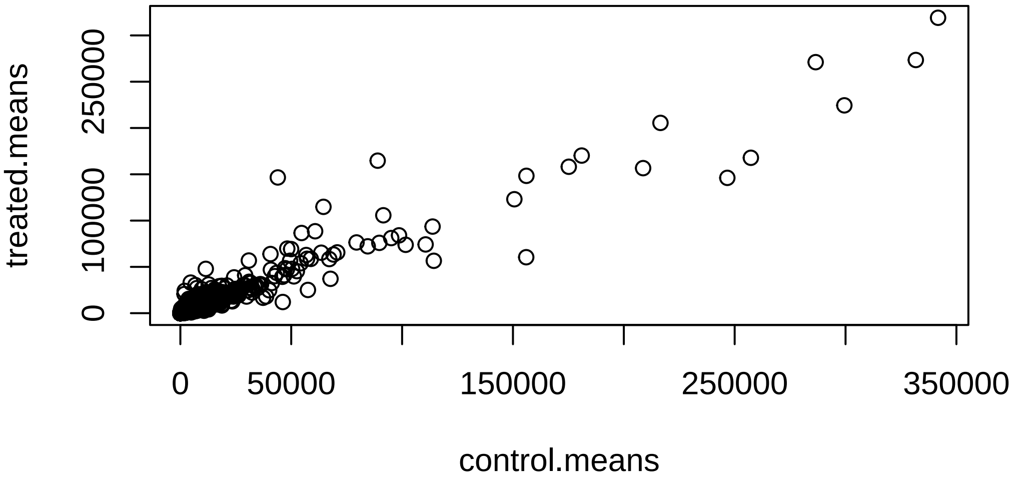
store these together for ease of book means counts

```
#5.
meancounts <- data.frame(control.means, treated.means)
head(meancounts)
```

```
                control.means treated.means
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
```

mamke a plot onf ocntorl vs treatments menad valeues for all genes

```
plot(meancounts)
```

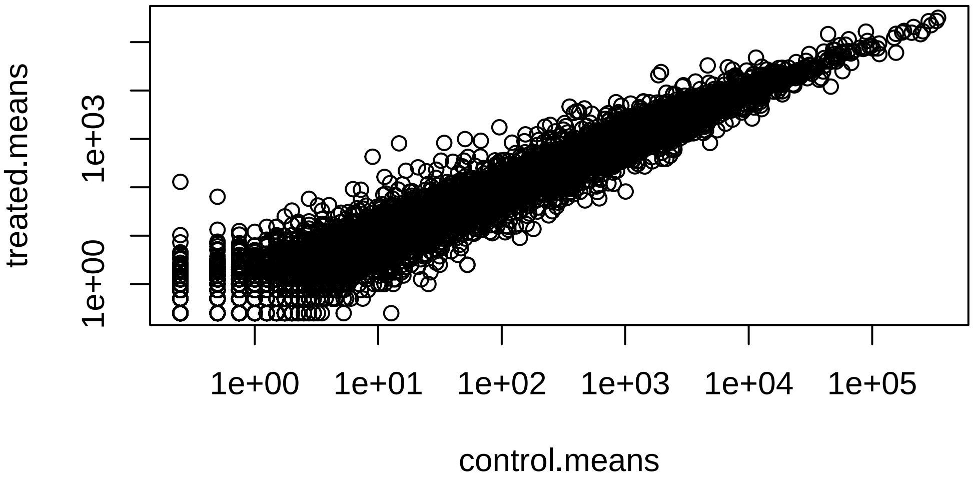


Make this a log plot

```
plot(meancounts,log="xy")
```

```
Warning in xy.coords(x, y, xlabel, ylabel, log): 15032 x values <= 0 omitted
from logarithmic plot
```

```
Warning in xy.coords(x, y, xlabel, ylabel, log): 15281 y values <= 0 omitted
from logarithmic plot
```



we often talk about metrics like "log2 fold-change"

```
#treated/control
log2(10/10)
```

```
[1] 0
```

```
log2(10/20)
```

```
[1] -1
```

```
log2(20/10)
```

```
[1] 1
```

```
log2(40/10)
```

```
[1] 2
```

```
log2(10/40)
```

```
[1] -2
```

lets calculate the log2 fold change four our treated over control mean counts

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

```
head(meancounts)
```

```
                control.means treated.means      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
```

A common "rule of thumb" is a log2 fold change cutoff of +2 and -2 to call genes "up regulated" or "down regulated"

Number of "up" genes

```
sum(meancounts$log2fc >= +2, na.rm=T)
```

```
[1] 1910
```

number of "down" genes at -2 threshold

```
sum(meancounts$log2fc <= -2, na.rm=T)
```

```
[1] 2330
```

##DESeq2 analysis

lest do this analysis properlty and keep our inner starts nerd happy are the teh diferrenes we seen and no drug significnat givne the replciate experimnet

```r
library(DESeq2)
```

```
Warning: package 'IRanges' was built under R version 4.4.2
```

```
Warning: package 'GenomeInfoDb' was built under R version 4.4.2
```

```
Warning: package 'MatrixGenerics' was built under R version 4.4.2
```

```
Warning: package 'matrixStats' was built under R version 4.4.3
```

for DESeq analysis we need thre things -count values (‘countData’) -metadat telling us about the columns in ‘counData’ (‘colData’) -design of the experiemnt (what do you want to compare)

our first function formDESeq 2 will set up the input required for analysis by storing all these 3 thigs together.

```r
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
```

the main function in DESeq2 thats runs the nalysis is called DESeq()

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

```
estimating size factors
```

```
estimating dispersions
```

```
gene-wise dispersion estimates
```

```
mean-dispersion relationship
```

```
final dispersion estimates
```

```
fitting model and testing
```

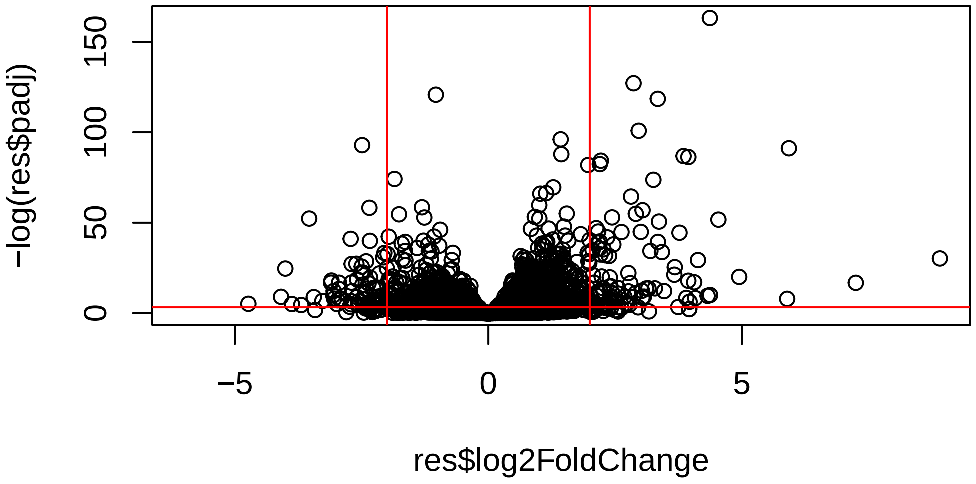
```
res<-results(dds)
head(res)
```

```
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
```

Volcano Plot

this us common summary result fomr figrue thse tyoes of expreimnts and plot the log2 fold change vs the adjusted p-value

```
plot(res$log2FoldChange,-log(res$padj))
abline(v=c(-2,2),col="red")
abline(h=-log(0.04),col="red")
```



save our results

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

add gene anotation

to help us make sense the results and comunicate them to other folks we need to add some more annotation to our main `res` object

we will use two bioconductor packages to first map IDS to different formats including teh clasic gene “symbol” gene name.

`BiocManager::install("AnnotationDbi")` ‘BiocManager::install(“org.Hs.eg.db”)

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

let's see what is in `org.Hs.eg.db` with the `columns()` function:

```
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"
```

We can translate or "map" IDs between any of these 26 database using the `mapIds()` function.

```
res$symbol<-mapIds(keys=row.names (res), #our current file
       keytype = "ENSEMBL", #the format of our Ids
       x= org.Hs.eg.db,         #Where to get the mapping from
       column = "SYMBOL")    # the format/DB to map to
```

```
'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 the mapping for “GENENAME” AND “ENTREZID” and store as `res$genenome` and `res$entrez`

```
res$entrez<-mapIds(keys=row.names (res),
       keytype = "ENSEMBL",
       x= org.Hs.eg.db,
       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 8 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      entrez
                <numeric> <character> <character>
ENSG00000000003  0.163035      TSPAN6        7105
ENSG00000000005        NA        TNMD       64102
ENSG00000000419  0.176032        DPM1        8813
ENSG00000000457  0.961694       SCYL3       57147
ENSG00000000460  0.815849       FIRRM       55732
ENSG00000000938        NA         FGR        2268
```

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

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

```r
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      entrez               genename
                <numeric> <character> <character>            <character>
ENSG00000000003  0.163035      TSPAN6        7105          tetraspanin 6
ENSG00000000005        NA        TNMD       64102            tenomodulin
ENSG00000000419  0.176032        DPM1        8813 dolichyl-phosphate m..
ENSG00000000457  0.961694       SCYL3       57147 SCY1 like pseudokina..
ENSG00000000460  0.815849       FIRRM       55732 FIGNL1 interacting r..
ENSG00000000938        NA         FGR        2268 FGR proto-oncogene, ..
```

Pathway Analysis

there are lots of bioconductor packages to do this type of analysis. for now lets just try one called **gage** again we need to install thid if we dont have it already.

```r
library(gage)
library(gageData)
library(pathview)
```

To use **gage** I need two things - a name vector of fold- change values for ourDEGs (our geneset of interest) - a set of pathways or genesets to use for annotation.

```r
x<-c(5,10)
```

```r
names(x)<-c("low","high")
x
```

```
 low high 
   5   10 
```

```r
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
```

```r
data(kegg.sets.hs)

keggres = gage(foldchanges, gsets=kegg.sets.hs)
```

In our results object we have:

```r
attributes(keggres)
```

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

```r
head(keggres$less,5)
```

```
                                                         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
                                                             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
                                                      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
```

lets look at one of these pathways with our genes colored up so we can see the overlap.

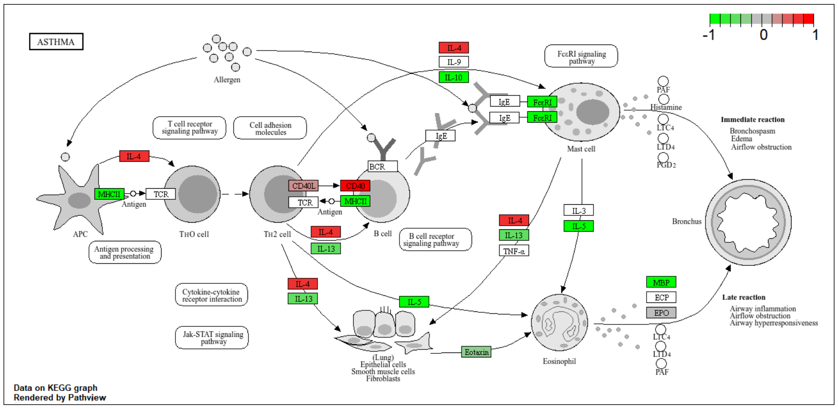
```
pathview(pathway.id = "hsa05310",gene.data=foldchanges)
```

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

```
Info: Working in directory C:/Users/rache/Documents/BIMM 143/Class 12
```

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

Add this pathway figure to our lab report:



Save our main results

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