# class12

## Section 1. Differential Expression Analysis

```
#Load Packages
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, 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
```

```
library(ggplot2)
library(org.Hs.eg.db)
```

```
## Loading required package: AnnotationDbi

##
```

```
library(AnnotationDbi)
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).
## ##############################################################################
```

```
library(gage)
```

```
##
```

```
library(gageData)
```

## Data Formatting

```
#Load Data Files
colData = read.csv("GSE37704_metadata.csv") #metadata file
countData = read.csv("GSE37704_featurecounts.csv", row.names=1) #counts file
```

Length column in countData is not relevant for further processing… Get colData and countData to match in format [DO NOT REPEAT - will mess up code]

```
#colData <- t(colData) #DO NOT transpose data to match countData <-> inverse x & y are OK!!!
countData <- countData[,-1] #Remove column `length` because colData doesn't have reference to it
```

Log transformations will not work on zero values, and genes with zeros across all cells

```
#Remove zero values from countData
countData <- countData[rowSums(countData) != 0, ]
```

```
#Put Data together in DESeq2 format
dds = DESeqDataSetFromMatrix(countData=countData,
                             colData=colData,
                             design=~condition)
```

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

## Running DESeq2

```
#Run the DESeq2 pipeline
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, contrast=c("condition", "hoxa1_kd", "control_sirna"))

#Visualize results
summary(res) #4349 genes up-regulated (27%) #4396 genes down-regulated (28%)
```

```
##
## out of 15975 with nonzero total read count
## adjusted p-value < 0.1
## LFC > 0 (up)       : 4349, 27%
## LFC < 0 (down)     : 4396, 28%
## outliers [1]       : 0, 0%
## low counts [2]     : 1237, 7.7%
## (mean count < 0)
## [1] see ’cooksCutoff’ argument of ?results
## [2] see ’independentFiltering’ argument of ?results
```

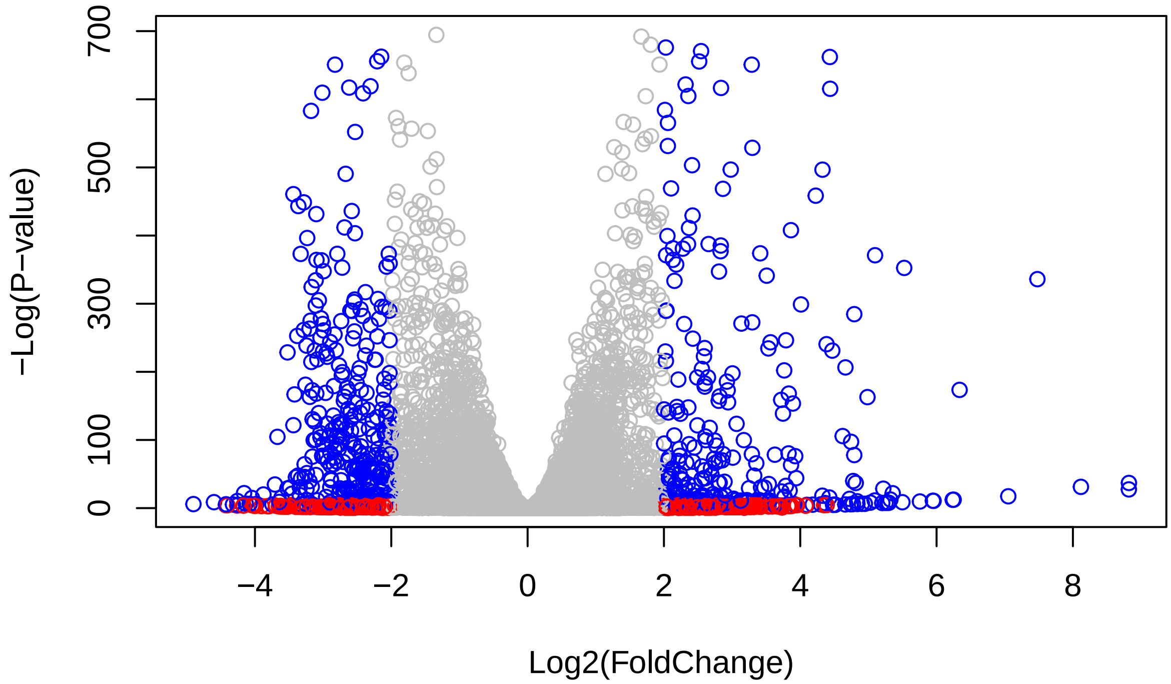
## Volcano Plot

To visualize the results, we can make a volcano plot. We will color the points based on their relevance: grey = default, red = only fold change > 2, blue = adj p-value < 0.01 && fold change > 2. The colorings proceed in the order from least stringent requirements to most stringent requirements because each step colors over points that already have a color.

```
#Base plot code
#plot( res$log2FoldChange, -log(res$padj) )

# Make a color vector for all genes; same length as the number of results points
volcano.colors <- rep("gray", nrow(res) )
volcano.colors[ abs(res$log2FoldChange) > 2 ] <- "red" #color corresponding points blue
volcano.colors[ (res$padj < 0.01) & (abs(res$log2FoldChange) > 2 ) ] <- "blue" #color corresponding poi

#Replot the plot
plot( res$log2FoldChange, -log(res$padj), col=volcano.colors, xlab="Log2(FoldChange)", ylab="-Log(P-valu
```



## Adding Gene Annotation

KEGG Pathway Analysis uses Entrez IDs, not Ensembl IDs, so we need to add a traslation column to res (DESeq2 results file)

```
#Add Entrez Gene IDs as a column in res
res$Entrez <- mapIds(org.Hs.eg.db,
                     keys = row.names(res),
                     keytype = "ENSEMBL",
                     column = "ENTREZID",
                     multiVals = "first")
```

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

```
#Add Gene Symbols as a column in res
res$Symbol <- mapIds(org.Hs.eg.db,
                     keys = row.names(res),
                     keytype = "ENSEMBL",
                     column = "SYMBOL",
                     multiVals = "first")
```

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

```
#Add Gene Names as a column in res
res$Name <- mapIds(org.Hs.eg.db,
                     keys = row.names(res),
                     keytype = "ENSEMBL",
                     column = "GENENAME",
                     multiVals = "first")
```

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

```
# Reorder results by adjusted p-value and SAVE to a CSV file.
write.csv(res[order(res$pvalue),], file="DESeq_results.csv")
```

## Section 2. Pathway Analysis

Here we are going to use the gage package for pathway analysis. Once we have a list of enriched pathways, we're going to use the pathview package to draw pathway diagrams, shading the molecules in the pathway by their degree of up/down-regulation.

### KEGG Pathways

```

data(kegg.sets.hs)
data(sigmet.idx.hs)

# Select for signaling and metabolic pathways only
kegg.sets.hs = kegg.sets.hs[sigmet.idx.hs]
```

The main `gage()` function requires a named vector of fold changes, where the names of the values are the Entrez gene IDs.

```
#Make the data vector first
gage.folds <- res$log2FoldChange

#Assign names to vector values
names(gage.folds) <- res$Entrez
```

```
#Run the gage pathway analysis pipeline
keggres = gage(gage.folds, gsets=kegg.sets.hs, same.dir=TRUE)
```

```
#Saves as a PNG
pathview(gene.data=gage.folds, pathway.id="hsa04110")
```

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

```
## Info: Working in directory C:/Users/rodeo/Documents/School/BIMM 143/R Projects/Class12 - RNA-Seq Anal
```

```
## Info: Writing image file hsa04110.pathview.png
```

```
#Saves as a PDF
pathview(gene.data=gage.folds, pathway.id="hsa04110", kegg.native=FALSE)
```

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

## Info: Working in directory C:/Users/rodeo/Documents/School/BIMM 143/R Projects/Class12 - RNA-Seq Anal

## Info: Writing image file hsa04110.pathview.pdf
```

### Top Five

```
## Focus on top 5 up-regulated pathways here for demo purposes only
keggrespathways <- rownames(keggres$greater)[1:5]

# Extract the 8 character long IDs part of each string
keggresids = substr(keggrespathways, start=1, stop=8)
keggresids
```

```
## [1] "hsa04640" "hsa04630" "hsa00140" "hsa04142" "hsa04330"
```

```
#Visualize with pathview()
pathview(gene.data=gage.folds, pathway.id=keggresids, species="hsa")
```

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

## Info: Working in directory C:/Users/rodeo/Documents/School/BIMM 143/R Projects/Class12 - RNA-Seq Anal

## Info: Writing image file hsa04640.pathview.png

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

## Info: Working in directory C:/Users/rodeo/Documents/School/BIMM 143/R Projects/Class12 - RNA-Seq Anal

## Info: Writing image file hsa04630.pathview.png

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

## Info: Working in directory C:/Users/rodeo/Documents/School/BIMM 143/R Projects/Class12 - RNA-Seq Anal

## Info: Writing image file hsa00140.pathview.png

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

## Info: Working in directory C:/Users/rodeo/Documents/School/BIMM 143/R Projects/Class12 - RNA-Seq Anal

## Info: Writing image file hsa04142.pathview.png

## Info: some node width is different from others, and hence adjusted!

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

## Info: Working in directory C:/Users/rodeo/Documents/School/BIMM 143/R Projects/Class12 - RNA-Seq Anal

## Info: Writing image file hsa04330.pathview.png
```

### Bottom Five

```
## Focus on top 5 up-regulated pathways here for demo purposes only
keggrespathways <- rownames(keggres$less)[1:5]

# Extract the 8 character long IDs part of each string
keggresids = substr(keggrespathways, start=1, stop=8)
keggresids
```

```
## [1] "hsa04110" "hsa03030" "hsa03013" "hsa03440" "hsa04114"
```

```
#Visualize with pathview()
pathview(gene.data=gage.folds, pathway.id=keggresids, species="hsa")
```

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

## Info: Working in directory C:/Users/rodeo/Documents/School/BIMM 143/R Projects/Class12 - RNA-Seq Ana

## Info: Writing image file hsa04110.pathview.png

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

## Info: Working in directory C:/Users/rodeo/Documents/School/BIMM 143/R Projects/Class12 - RNA-Seq Ana

## Info: Writing image file hsa03030.pathview.png

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

## Info: Working in directory C:/Users/rodeo/Documents/School/BIMM 143/R Projects/Class12 - RNA-Seq Ana

## Info: Writing image file hsa03013.pathview.png

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

## Info: Working in directory C:/Users/rodeo/Documents/School/BIMM 143/R Projects/Class12 - RNA-Seq Ana

## Info: Writing image file hsa03440.pathview.png

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

## Info: Working in directory C:/Users/rodeo/Documents/School/BIMM 143/R Projects/Class12 - RNA-Seq Ana

## Info: Writing image file hsa04114.pathview.png
```

> Q5. Can you do the same procedure as above to plot the pathview figures for the top 5 down-reguled pathways? ^ Done. (preceeding code chunk)

## Section 3. Gene Ontology (GO)

We can also do a similar procedure with gene ontology. Similar to above, go.sets.hs has all GO terms. go.subs.hs is a named list containing indexes for the BP, CC, and MF ontologies. Ontologies will tell us what specific processes are being dysregulated.

```
data(go.sets.hs)
data(go.subs.hs)

#Select for biological processes only
gobpsets = go.sets.hs[go.subs.hs$BP]

gobpres = gage(gage.folds, gsets=gobpsets, same.dir=TRUE)

lapply(gobpres, head)
```

```
## $greater
##                                             p.geomean stat.mean        p.val
## GO:0007156 homophilic cell adhesion       8.519724e-05  3.824205 8.519724e-05
## GO:0002009 morphogenesis of an epithelium 1.396681e-04  3.653886 1.396681e-04
## GO:0048729 tissue morphogenesis           1.432451e-04  3.643242 1.432451e-04
## GO:0007610 behavior                       2.195494e-04  3.530241 2.195494e-04
## GO:0060562 epithelial tube morphogenesis  5.932837e-04  3.261376 5.932837e-04
## GO:0035295 tube development               5.953254e-04  3.253665 5.953254e-04
##                                              q.val set.size         exp1
## GO:0007156 homophilic cell adhesion       0.1951953      113 8.519724e-05
## GO:0002009 morphogenesis of an epithelium 0.1951953      339 1.396681e-04
## GO:0048729 tissue morphogenesis           0.1951953      424 1.432451e-04
## GO:0007610 behavior                       0.2243795      427 2.195494e-04
## GO:0060562 epithelial tube morphogenesis  0.3711390      257 5.932837e-04
## GO:0035295 tube development               0.3711390      391 5.953254e-04
##
## $less
##                                             p.geomean stat.mean        p.val
## GO:0048285 organelle fission              1.536227e-15 -8.063910 1.536227e-15
## GO:0000280 nuclear division               4.286961e-15 -7.939217 4.286961e-15
## GO:0007067 mitosis                        4.286961e-15 -7.939217 4.286961e-15
## GO:0000087 M phase of mitotic cell cycle 1.169934e-14 -7.797496 1.169934e-14
## GO:0007059 chromosome segregation         2.028624e-11 -6.878340 2.028624e-11
## GO:0000236 mitotic prometaphase           1.729553e-10 -6.695966 1.729553e-10
##                                                 q.val set.size         exp1
## GO:0048285 organelle fission              5.841698e-12      376 1.536227e-15
## GO:0000280 nuclear division               5.841698e-12      352 4.286961e-15
## GO:0007067 mitosis                        5.841698e-12      352 4.286961e-15
## GO:0000087 M phase of mitotic cell cycle 1.195672e-11      362 1.169934e-14
## GO:0007059 chromosome segregation         1.658603e-08      142 2.028624e-11
## GO:0000236 mitotic prometaphase           1.178402e-07       84 1.729553e-10
##
## $stats
##                                           stat.mean     exp1
## GO:0007156 homophilic cell adhesion        3.824205 3.824205
## GO:0002009 morphogenesis of an epithelium  3.653886 3.653886
## GO:0048729 tissue morphogenesis            3.643242 3.643242
## GO:0007610 behavior                           3.530241 3.530241
## GO:0060562 epithelial tube morphogenesis   3.261376 3.261376
## GO:0035295 tube development                 3.253665 3.253665
```

## Section 4. Reactome Analysis

Reactome is database consisting of biological molecules and their relation to pathways and processes.

```
sig.genes <- res[res$padj <= 0.05 & !is.na(res$padj), "Symbol"]
print(paste("Total number of significant genes:", length(sig.genes)))
```

```
## [1] "Total number of significant genes: 8147"
```

```
#Save data
write.table(sig.genes, file="significant_genes.txt", row.names=FALSE, col.names=FALSE, quote=FALSE)
```

> Q6. The Endosomal/Vacuolar pathway has the most significant Entities p-value. No, the most significant pathways listed do not match previous KEGG results. There seems to not be anything in the code selecting for the highest/lowest expressing results from the gene ontology database.