Class 12: RNA-Seq Mini Project

Ramola Baviskar (PID A12228297)

2/24/2022

Here we'll work on a complete differential expression analysis project. We'll use DESeq2 for this.

```
library(DESeq2)
library(ggplot2)
library(org.Hs.eg.db)
library(AnnotationDbi)
library(pathview)
library(gage)
library(gageData)
```

#Step 1: Input the counts & metadata files.

```
countData <- read.csv("GSE37704_featurecounts.csv", row.names = 1)
colData <- read.csv("GSE37704_metadata.csv")
```

```
colData
```

```
##          id     condition
## 1 SRR493366 control_sirna
## 2 SRR493367 control_sirna
## 3 SRR493368 control_sirna
## 4 SRR493369      hoxa1_kd
## 5 SRR493370      hoxa1_kd
## 6 SRR493371      hoxa1_kd
```

```
countData <- countData[,-1]
head(countData[,-1])
```

```
##                 SRR493367 SRR493368 SRR493369 SRR493370 SRR493371
## ENSG00000186092         0         0         0         0         0
## ENSG00000279928         0         0         0         0         0
## ENSG00000279457        28        29        29        28        46
## ENSG00000278566         0         0         0         0         0
## ENSG00000273547         0         0         0         0         0
## ENSG00000187634       123       205       207       212       258
```

```
colData$id
```

```
## [1] "SRR493366" "SRR493367" "SRR493368" "SRR493369" "SRR493370" "SRR493371"
```

```
colnames(countData)
```

```
## [1] "SRR493366" "SRR493367" "SRR493368" "SRR493369" "SRR493370" "SRR493371"
```

```
all(colData$id == colnames(countData))
```

```
## [1] TRUE
```

Q. Complete the code below to filter countData to exclude genes (i.e. rows) where we have 0 read count across all samples (i.e. columns).

```
head(countData)
```

```
##                 SRR493366 SRR493367 SRR493368 SRR493369 SRR493370 SRR493371
## ENSG00000186092         0         0         0         0         0         0
## ENSG00000279928         0         0         0         0         0         0
## ENSG00000279457        23        28        29        29        28        46
## ENSG00000278566         0         0         0         0         0         0
## ENSG00000273547         0         0         0         0         0         0
## ENSG00000187634       124       123       205       207       212       258
```

```
counts <- countData [rowSums(countData) != 0,]
head(counts)
```

```
##                 SRR493366 SRR493367 SRR493368 SRR493369 SRR493370 SRR493371
## ENSG00000279457        23        28        29        29        28        46
## ENSG00000187634       124       123       205       207       212       258
## ENSG00000188976      1637      1831      2383      1226      1326      1504
## ENSG00000187961       120       153       180       236       255       357
## ENSG00000187583        24        48        65        44        48        64
## ENSG00000187642         4         9        16        14        16        16
```

#Step 2: Run DESeq The steps here are to first set up the object required by DESeq using the `DESeqDataSetFromMatrix()` function. This will store the counts and metadata along w/ the design of the experiment (ie where in the metadata we have the description of what the columns of counts correspond to.) ‘

```
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
```

Now I can run my differential expression w/ `DESeq()`

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

```
## estimating size factors
## estimating dispersions
```

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

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

```
## final dispersion estimates
```

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

Now get my results from this.

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

```
## log2 fold change (MLE): condition hoxa1 kd vs control sirna
## Wald test p-value: condition hoxa1 kd vs control sirna
## DataFrame with 19808 rows and 6 columns
##                  baseMean log2FoldChange     lfcSE      stat      pvalue
##                 <numeric>      <numeric> <numeric> <numeric>   <numeric>
## ENSG00000186092    0.0000             NA        NA        NA          NA
## ENSG00000279928    0.0000             NA        NA        NA          NA
## ENSG00000279457   29.9136       0.179257  0.324822  0.551863    0.581042
## ENSG00000278566    0.0000             NA        NA        NA          NA
## ENSG00000273547    0.0000             NA        NA        NA          NA
## ...                   ...            ...       ...       ...         ...
## ENSG00000277856     0.000             NA        NA        NA          NA
## ENSG00000275063     0.000             NA        NA        NA          NA
## ENSG00000271254   181.596      -0.609667   0.14132  -4.31407 1.60276e-05
## ENSG00000277475     0.000             NA        NA        NA          NA
## ENSG00000268674     0.000             NA        NA        NA          NA
##                      padj
##                 <numeric>
## ENSG00000186092        NA
## ENSG00000279928        NA
## ENSG00000279457   0.68708
## ENSG00000278566        NA
## ENSG00000273547        NA
## ...                   ...
## ENSG00000277856        NA
## ENSG00000275063        NA
## ENSG00000271254 4.5414e-05
## ENSG00000277475        NA
## ENSG00000268674        NA
```

```
summary(res)
```

```
##
## out of 15975 with nonzero total read count
## adjusted p-value < 0.1
## LFC > 0 (up)       : 4349, 27%
## LFC < 0 (down)     : 4393, 27%
## outliers [1]       : 0, 0%
## low counts [2]     : 1221, 7.6%
## (mean count < 0)
## [1] see 'cooksCutoff' argument of ?results
## [2] see 'independentFiltering' argument of ?results
```

#Step 4: Add annotation Q. Use the mapIDs() function multiple times to add SYMBOL, ENTREZID and GENENAME annotation to our results by completing the code below.

```
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"
```

```
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
```

```
res$entrez <- mapIds(org.Hs.eg.db,
                     keys=row.names(res),
                     keytype="ENSEMBL",
                     column="ENTREZID",
                     multiVals="first")
```

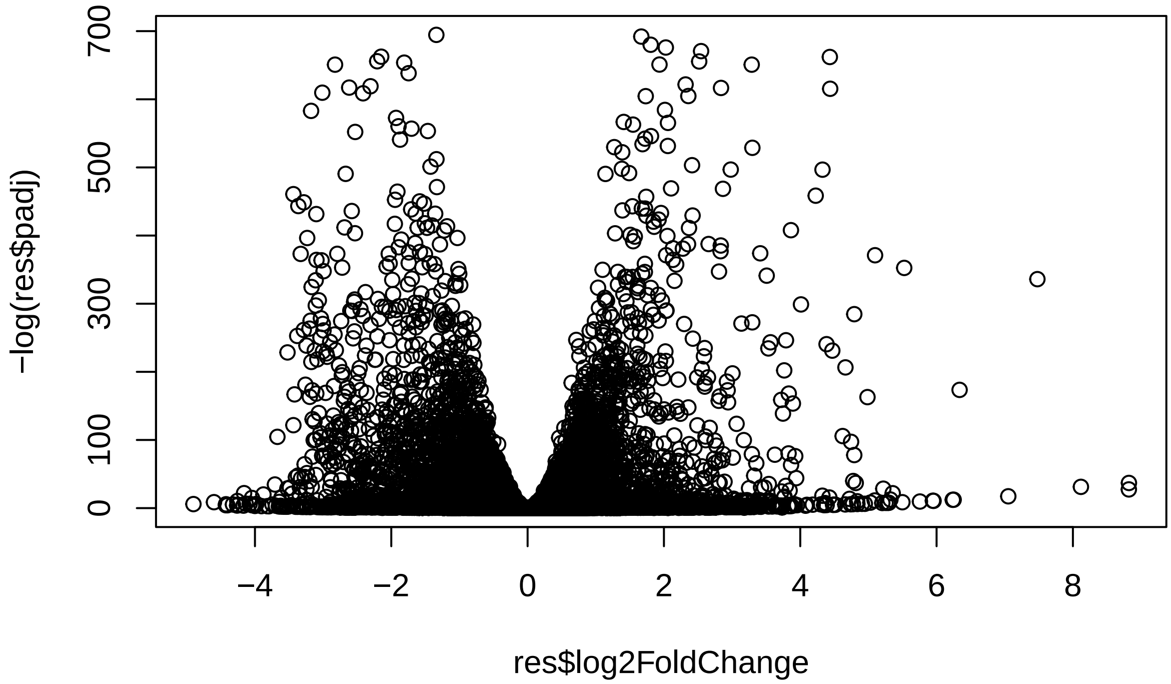
```
## 'select()' returned 1:many mapping between keys and columns
```

```
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
```

#Step 3: Volcano plot Common summary figure that gives a good overview of the results.

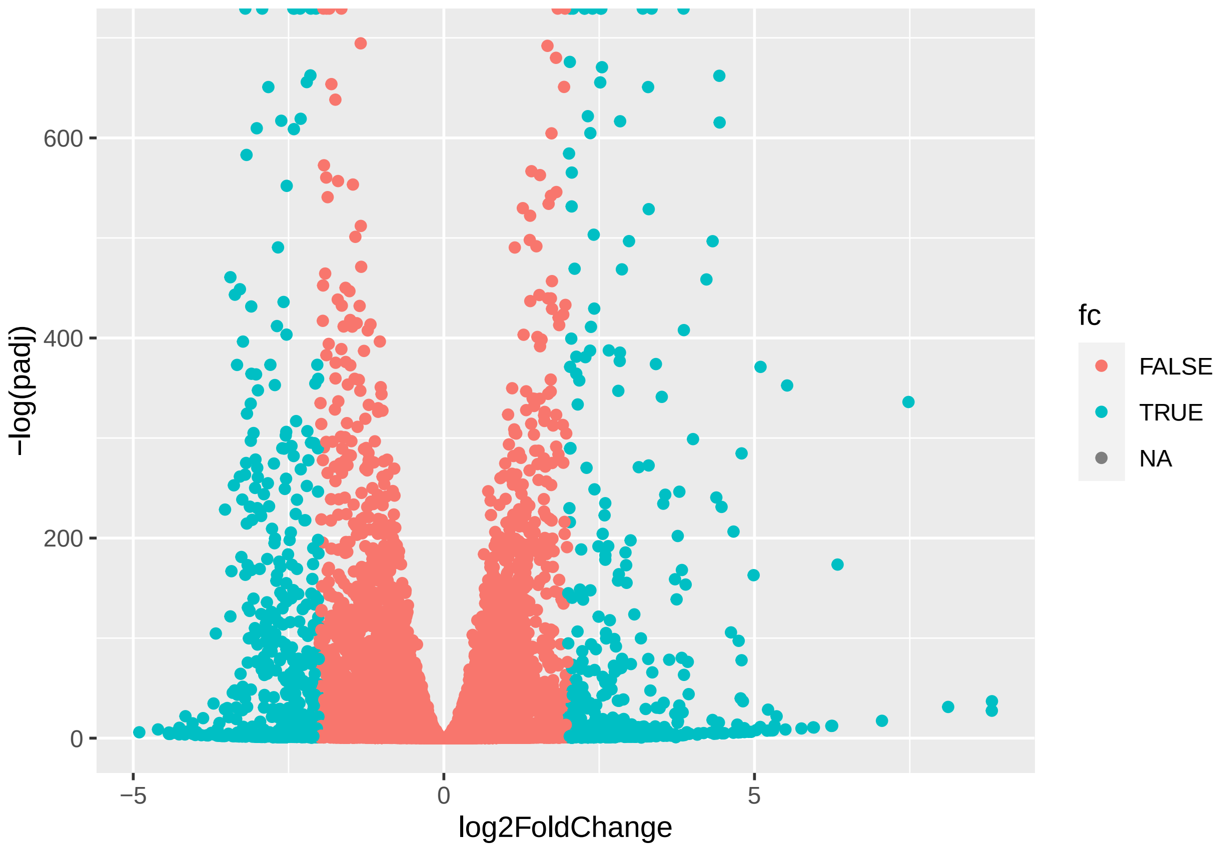
```
plot(res$log2FoldChange, -log(res$padj))
```



Try ggplot for this.

```
tmp <- as.data.frame(res)
tmp$fc <- abs(res$log2FoldChange) > 2
ggplot(tmp) +
  aes(log2FoldChange, -log(padj), col=fc) +
  geom_point()
```

```
## Warning: Removed 5054 rows containing missing values (geom_point).
```



```
BiocManager::install("EnhancedVolcano")
```

```
## Bioconductor version 3.14 (BiocManager 1.30.16), R 4.1.2 (2021-11-01)
```

```
## Warning: package(s) not installed when version(s) same as current; use ‘force = TRUE‘ to
##   re-install: 'EnhancedVolcano'
```

```
## Old packages: 'class', 'cli', 'colorspace', 'crayon', 'evaluate', 'foreign',
##   'glue', 'jsonlite', 'MASS', 'Matrix', 'mgcv', 'nlme', 'nnet', 'rpart',
##   'spatial', 'tidyselect', 'tinytex', 'XML', 'yaml'
```

```
library(EnhancedVolcano)
```

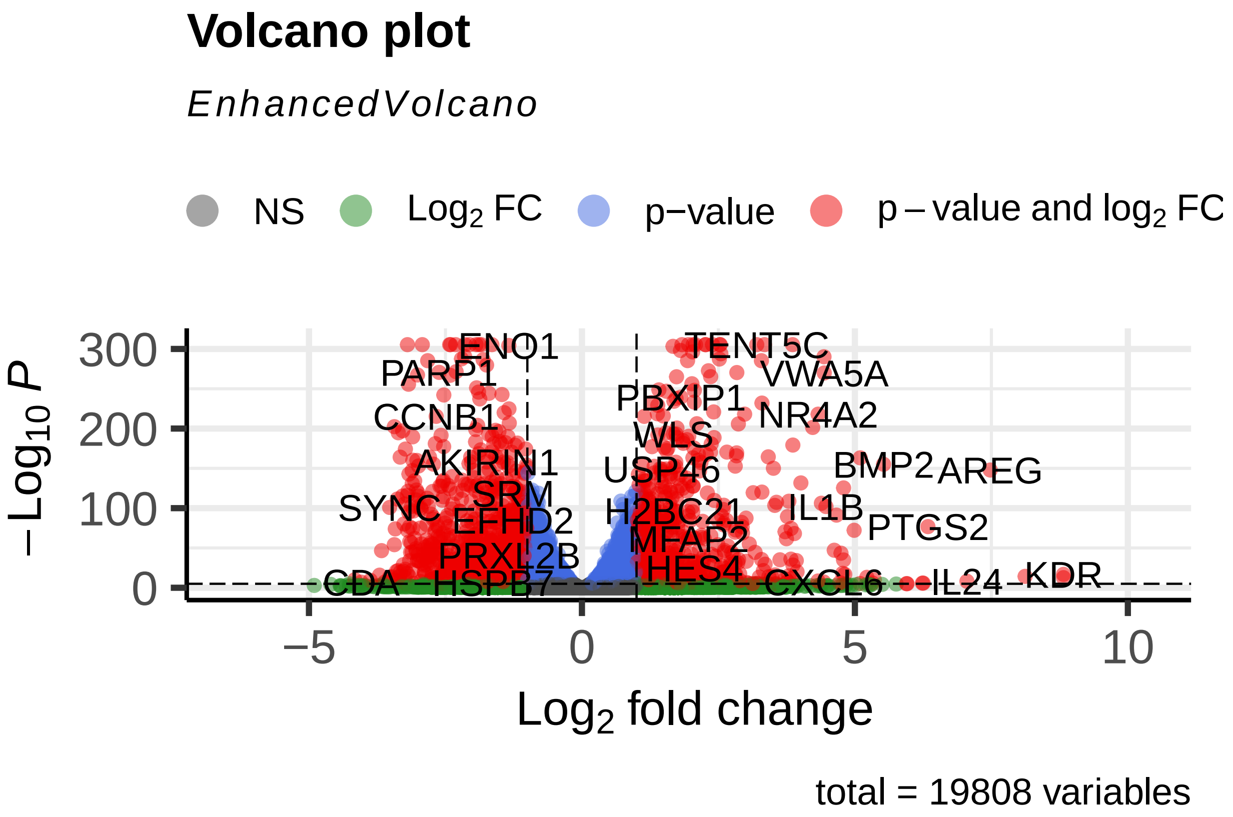
```
## Loading required package: ggrepel
```

```
## Registered S3 methods overwritten by 'ggalt':
##   method                  from
##   grid.draw.absoluteGrob  ggplot2
##   grobHeight.absoluteGrob ggplot2
##   grobWidth.absoluteGrob  ggplot2
##   grobX.absoluteGrob      ggplot2
##   grobY.absoluteGrob      ggplot2
```

```
x <- as.data.frame(res)

EnhancedVolcano(x,
    lab = x$symbol,
    x = 'log2FoldChange',
    y = 'pvalue')
```

```
## Warning: One or more p-values is 0. Converting to 10^-1 * current lowest non-
## zero p-value...
```



#Step 5: Pathway analysis

Here we try to bring back the biology and help with the interpretation of our results. We try to answer the question: which pathways and functions feature heavily in our differentially expressed genes? Recall that we need a "vector of iportance" as input for GAGE that has ENTREZ ids set as the names attribute.

```
foldchange <- res$log2FoldChange
names(foldchange) <- res$entrez
```

```
data(kegg.sets.hs)
data(sigmet.idx.hs)
kegg.sets.hs = kegg.sets.hs[sigmet.idx.hs]
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"
```

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

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

Look at the first 2 downregulated pathways.

```
head(keggres$less, 2)
```

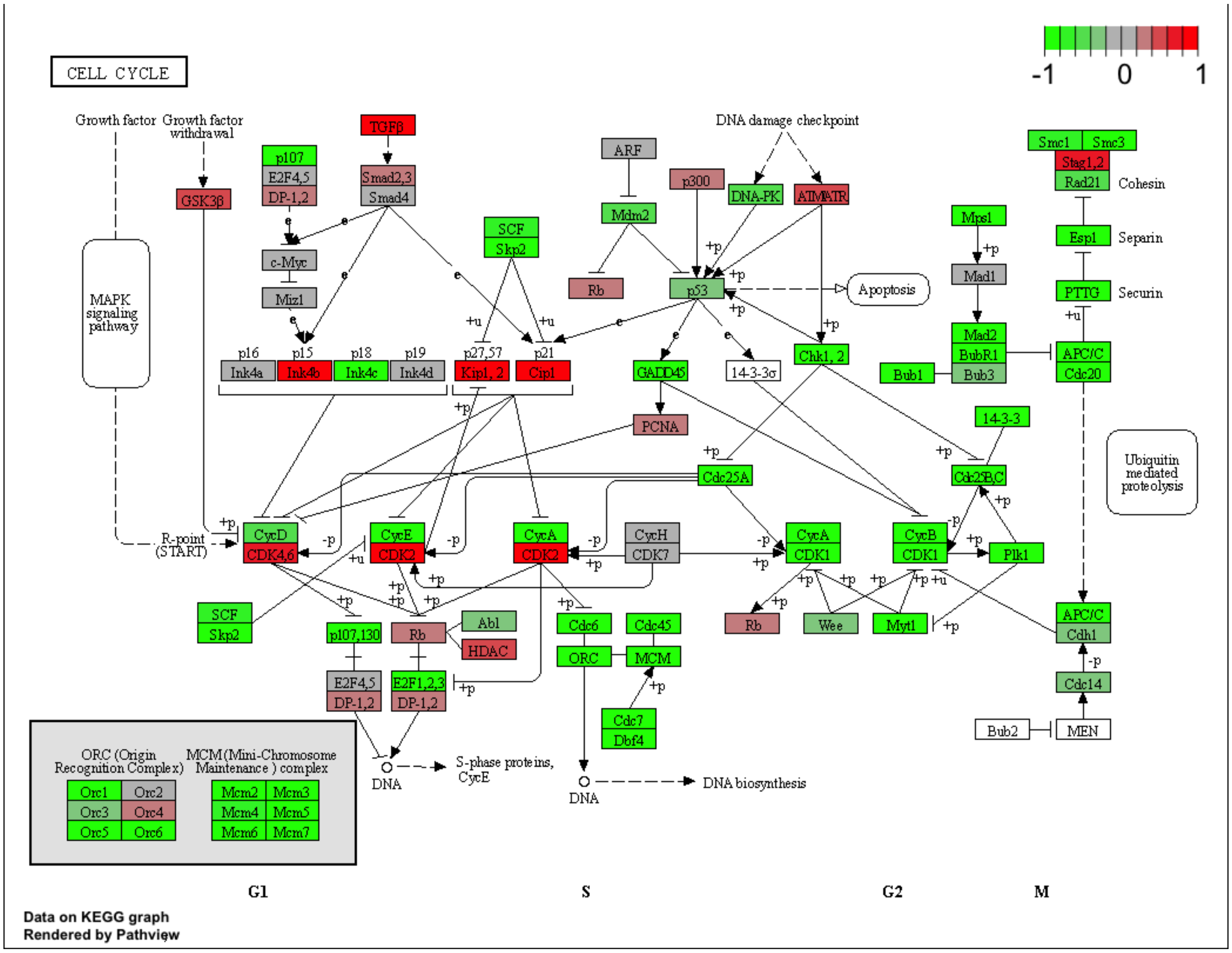
```
##                             p.geomean stat.mean        p.val       q.val
## hsa04110 Cell cycle      7.077982e-06 -4.432593 7.077982e-06 0.001160789
## hsa03030 DNA replication 9.424076e-05 -3.951803 9.424076e-05 0.007727742
##                          set.size         exp1
## hsa04110 Cell cycle           124 7.077982e-06
## hsa03030 DNA replication       36 9.424076e-05
```

```
pathview(foldchange, pathway.id="hsa04110")
```

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

```
## Info: Working in directory /Users/Ramola/Desktop/BIMM143/class12
```

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



##Gene Ontology analysis We can use a different gene set database (we used KEGG above) to provide different (but hopefully complementary) information. We will try GO here w/ a focus on Biological Pathways (BP) component of GO.

```
data(go.sets.hs)
data(go.subs.hs)
gobpsets = go.sets.hs[go.subs.hs$BP]

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

lapply(gobpres, head)
```

```
## $greater
##                                              p.geomean stat.mean        p.val
## GO:0007156 homophilic cell adhesion        1.624062e-05  4.226117 1.624062e-05
## GO:0048729 tissue morphogenesis            5.407952e-05  3.888470 5.407952e-05
## GO:0002009 morphogenesis of an epithelium  5.727599e-05  3.878706 5.727599e-05
## GO:0030855 epithelial cell differentiation 2.053700e-04  3.554776 2.053700e-04
## GO:0060562 epithelial tube morphogenesis   2.927804e-04  3.458463 2.927804e-04
## GO:0048598 embryonic morphogenesis         2.959270e-04  3.446527 2.959270e-04
##                                                 q.val set.size         exp1
## GO:0007156 homophilic cell adhesion        0.07103646      138 1.624062e-05
## GO:0048729 tissue morphogenesis            0.08350839      483 5.407952e-05
## GO:0002009 morphogenesis of an epithelium  0.08350839      382 5.727599e-05
## GO:0030855 epithelial cell differentiation 0.15370245      299 2.053700e-04
## GO:0060562 epithelial tube morphogenesis   0.15370245      289 2.927804e-04
## GO:0048598 embryonic morphogenesis         0.15370245      498 2.959270e-04
## 
## $less
##                                             p.geomean stat.mean        p.val
## GO:0048285 organelle fission             6.386337e-16 -8.175381 6.386337e-16
## GO:0000280 nuclear division              1.726380e-15 -8.056666 1.726380e-15
## GO:0007067 mitosis                       1.726380e-15 -8.056666 1.726380e-15
## GO:0000087 M phase of mitotic cell cycle 4.593581e-15 -7.919909 4.593581e-15
## GO:0007059 chromosome segregation        9.576332e-12 -6.994852 9.576332e-12
## GO:0051301 cell division                 8.718528e-11 -6.455491 8.718528e-11
##                                                 q.val set.size         exp1
## GO:0048285 organelle fission             2.517062e-12      386 6.386337e-16
## GO:0000280 nuclear division              2.517062e-12      362 1.726380e-15
## GO:0007067 mitosis                       2.517062e-12      362 1.726380e-15
## GO:0000087 M phase of mitotic cell cycle 5.023080e-12      373 4.593581e-15
## GO:0007059 chromosome segregation        8.377375e-09      146 9.576332e-12
## GO:0051301 cell division                 6.355807e-08      479 8.718528e-11
## 
## $stats
##                                             stat.mean     exp1
## GO:0007156 homophilic cell adhesion          4.226117 4.226117
## GO:0048729 tissue morphogenesis              3.888470 3.888470
## GO:0002009 morphogenesis of an epithelium    3.878706 3.878706
## GO:0030855 epithelial cell differentiation   3.554776 3.554776
## GO:0060562 epithelial tube morphogenesis     3.458463 3.458463
## GO:0048598 embryonic morphogenesis           3.446527 3.446527
```

```r
head(gobpres$less)
```

```
##                                             p.geomean stat.mean        p.val
## GO:0048285 organelle fission             6.386337e-16 -8.175381 6.386337e-16
## GO:0000280 nuclear division              1.726380e-15 -8.056666 1.726380e-15
## GO:0007067 mitosis                       1.726380e-15 -8.056666 1.726380e-15
## GO:0000087 M phase of mitotic cell cycle 4.593581e-15 -7.919909 4.593581e-15
## GO:0007059 chromosome segregation        9.576332e-12 -6.994852 9.576332e-12
## GO:0051301 cell division                 8.718528e-11 -6.455491 8.718528e-11
##                                                 q.val set.size         exp1
## GO:0048285 organelle fission             2.517062e-12      386 6.386337e-16
## GO:0000280 nuclear division              2.517062e-12      362 1.726380e-15
## GO:0007067 mitosis                       2.517062e-12      362 1.726380e-15
## GO:0000087 M phase of mitotic cell cycle 5.023080e-12      373 4.593581e-15
## GO:0007059 chromosome segregation        8.377375e-09      146 9.576332e-12
## GO:0051301 cell division                 6.355807e-08      479 8.718528e-11
```

##Reactome We can use Reactome either as an R package (like above) or we an use the website. The website needs a file of "gene important" just like gage above. Reactome is a database consisting of biological molecules and their relation to pathways and processes.

```
sig_genes <- res[res$padj <= 0.05 & !is.na(res$padj), "symbol"]
```

```
write.table(sig_genes, file="significant_genes.txt", row.names=FALSE, col.names=FALSE, quote=FALSE)
```

#Save my results

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