# Class14: RNAseq mini project

Hanhee Jo (PID# 59017994)

## Table of contents

## Background

The data for for hands-on session comes from GEO entry: GSE37704, which is associated with the following publication:

> Trapnell C, Hendrickson DG, Sauvageau M, Goff L et al. "Differential analysis of gene regulation at transcript resolution with RNA-seq". Nat Biotechnol 2013 Jan;31(1):46-53. PMID: 23222703

The authors report on differential analysis of lung fibroblasts in response to loss of the developmental transcription factor HOXA1.

## Data Import

Reading in the counts and the metadata

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

```
head(counts)
```

```
                length SRR493366 SRR493367 SRR493368 SRR493369 SRR493370
ENSG00000186092    918         0         0         0         0         0
ENSG00000279928    718         0         0         0         0         0
ENSG00000279457   1982        23        28        29        29        28
ENSG00000278566    939         0         0         0         0         0
ENSG00000273547    939         0         0         0         0         0
ENSG00000187634   3214       124       123       205       207       212
                SRR493371
ENSG00000186092         0
ENSG00000279928         0
ENSG00000279457        46
ENSG00000278566         0
ENSG00000273547         0
ENSG00000187634       258
```

```
metadata
```

```
         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
```

## Tidy and verify data

> Q. How many genes are in this dataset?

```r
nrow(counts)
```

```
[1] 19808
```

> Q. How many control and knockdown experiments are there?

```r
table (metadata$condition)
```

```

control_sirna      hoxa1_kd 
            3             3 
```

> Q. Does the `metadata` match to the `countdata`

```r
all (colnames(counts) == metadata$id)
```

```
Warning in colnames(counts) == metadata$id: longer object length is not a
multiple of shorter object length
```

```
[1] FALSE
```

```r
colnames(counts)
```

```
[1] "length"    "SRR493366" "SRR493367" "SRR493368" "SRR493369" "SRR493370"
[7] "SRR493371"
```

```r
metadata$id
```

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

###Fix countdata to match the coldata/metadata

```r
newcounts <- counts[, -1]
dim(newcounts)
```

```
[1] 19808     6
```

```
all (colnames(newcounts) == metadata$id)
```

```
[1] TRUE
```

### Remove zero count genes

```
to.keep <- rowSums(newcounts) != 0
countData <- newcounts[to.keep, ]
head(countData)
```

```
                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
```

## PCA quality control

We can use `prcomp()` function.

```
pc <- prcomp( t(countData), scale = T)
summary(pc)
```

```
Importance of components:
                           PC1     PC2      PC3      PC4      PC5       PC6
Standard deviation     87.7211 73.3196 32.89604 31.15094 29.18417 7.387e-13
Proportion of Variance  0.4817  0.3365  0.06774  0.06074  0.05332 0.000e+00
Cumulative Proportion   0.4817  0.8182  0.88594  0.94668  1.00000 1.000e+00
```

```
metadata$condition
```

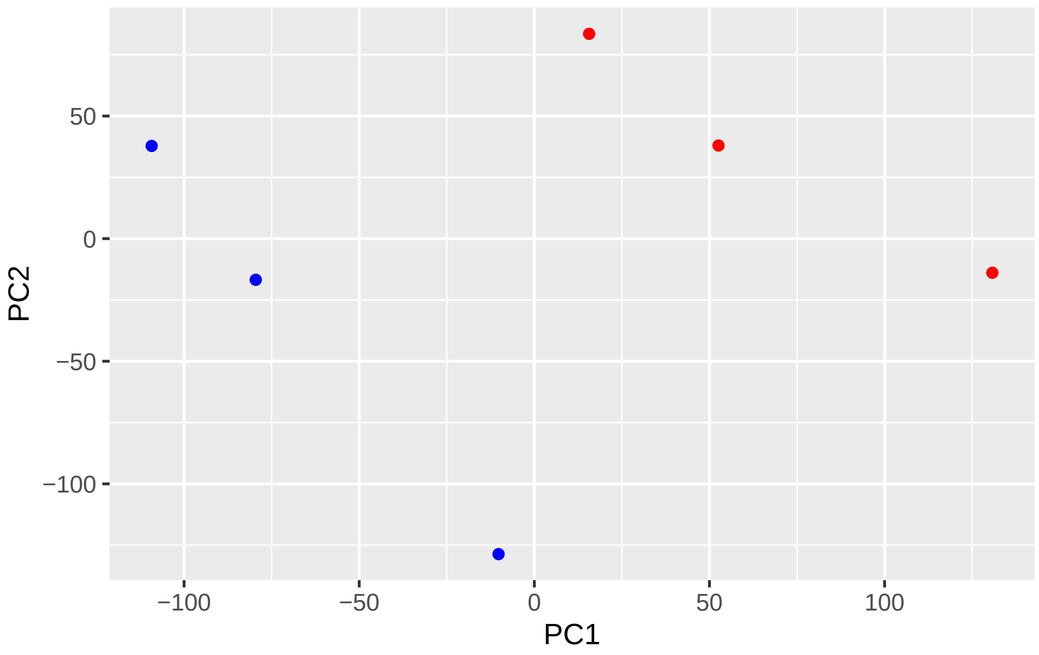
```
[1] "control_sirna" "control_sirna" "control_sirna" "hoxa1_kd"
[5] "hoxa1_kd"      "hoxa1_kd"
```

```
mycols <- c( rep("blue", 3), rep("red", 3) )
```

Color by “control” (blue) or “kd” (red)

```
library(ggplot2)

ggplot(pc$x) +
  aes(PC1, PC2) +
  geom_point(col = mycols)
```



Q. How many genes do we have left after filtering?

```
nrow(countData)
```

```
[1] 15975
```

## DESeq analysis

```r
library(DESeq2)
```

```
Warning: package 'DESeq2' was built under R version 4.3.3
```

```
Warning: package 'S4Vectors' was built under R version 4.3.2
```

```
Warning: package 'GenomeInfoDb' was built under R version 4.3.3
```

```
Warning: package 'SummarizedExperiment' was built under R version 4.3.2
```

```
Warning: package 'matrixStats' was built under R version 4.3.3
```

### Setup the DESeq input object

```r
dds <- DESeqDataSetFromMatrix(countData = countData,
                              colData = metadata,
                              design = ~condition)
```

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

### Run DESeq

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

```
estimating size factors
```

```
estimating dispersions
```

```
gene-wise dispersion estimates
```

```
mean-dispersion relationship
```

```
final dispersion estimates
```

```
fitting model and testing
```

### Extract results

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

```
head(res)
```

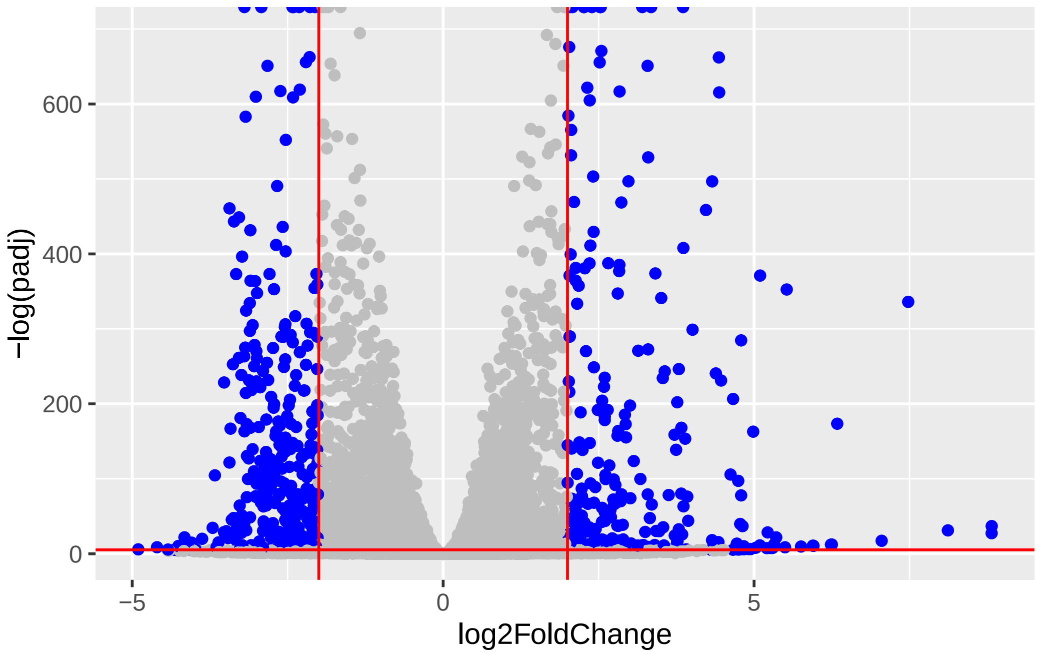
```
log2 fold change (MLE): condition hoxa1 kd vs control sirna
Wald test p-value: condition hoxa1 kd vs control sirna
DataFrame with 6 rows and 6 columns
                 baseMean log2FoldChange     lfcSE       stat      pvalue
                <numeric>      <numeric> <numeric>  <numeric>   <numeric>
ENSG00000279457   29.9136      0.1792571 0.3248216   0.551863 5.81042e-01
ENSG00000187634  183.2296      0.4264571 0.1402658   3.040350 2.36304e-03
ENSG00000188976 1651.1881     -0.6927205 0.0548465 -12.630158 1.43989e-36
ENSG00000187961  209.6379      0.7297556 0.1318599   5.534326 3.12428e-08
ENSG00000187583   47.2551      0.0405765 0.2718928   0.149237 8.81366e-01
ENSG00000187642   11.9798      0.5428105 0.5215599   1.040744 2.97994e-01
                       padj
                  <numeric>
ENSG00000279457 6.86555e-01
ENSG00000187634 5.15718e-03
ENSG00000188976 1.76549e-35
ENSG00000187961 1.13413e-07
ENSG00000187583 9.19031e-01
ENSG00000187642 4.03379e-01
```

## Volcano plot

```
mycols <- rep("gray", nrow(res))
mycols[ abs(res$log2FoldChange) > 2 ]  <- "blue"
mycols[ res$padj > 0.005] <- "gray"
```

```
ggplot(res) +
  aes(log2FoldChange, -log(padj)) +
  geom_point(col=mycols) +
  geom_vline(xintercept = c(-2, 2), col="red") +
  geom_hline(yintercept = -log(0.005), col="red")
```

```
Warning: Removed 1237 rows containing missing values or values outside the scale range
(`geom_point()`).
```



## Add gene annotation

We want to add gene SYMBOL and ENTREZID values to our results object.

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

```
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"
```

```r
res$symbol <- mapIds(org.Hs.eg.db,
                     keys=rownames(res), # Our genenames
                     keytype="ENSEMBL",   # The format of our genenames
                     column="SYMBOL",     # The new format we want to add
                     multiVals="first")
```

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

```r
res$entrez<- mapIds(org.Hs.eg.db,
                     keys=rownames(res),
                     keytype="ENSEMBL",
                     column="ENTREZID", # The new format we want
                     multiVals="first")
```

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

```r
head(res)
```

```
log2 fold change (MLE): condition hoxa1 kd vs control sirna
Wald test p-value: condition hoxa1 kd vs control sirna
DataFrame with 6 rows and 8 columns
                 baseMean log2FoldChange     lfcSE      stat      pvalue
                <numeric>      <numeric> <numeric> <numeric>   <numeric>
ENSG00000279457   29.9136      0.1792571 0.3248216   0.551863 5.81042e-01
ENSG00000187634  183.2296      0.4264571 0.1402658   3.040350 2.36304e-03
ENSG00000188976 1651.1881     -0.6927205 0.0548465 -12.630158 1.43989e-36
ENSG00000187961  209.6379      0.7297556 0.1318599   5.534326 3.12428e-08
ENSG00000187583   47.2551      0.0405765 0.2718928   0.149237 8.81366e-01
ENSG00000187642   11.9798      0.5428105 0.5215599   1.040744 2.97994e-01
                       padj      symbol      entrez
                  <numeric> <character> <character>
ENSG00000279457 6.86555e-01          NA          NA
ENSG00000187634 5.15718e-03      SAMD11      148398
ENSG00000188976 1.76549e-35       NOC2L       26155
ENSG00000187961 1.13413e-07      KLHL17      339451
ENSG00000187583 9.19031e-01     PLEKHN1       84069
ENSG00000187642 4.03379e-01       PERM1       84808
```

## Save results

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

## Pathway analysis

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

### KEGG

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

```
head(kegg.sets.hs, 1)
```

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

Make an input vector for `gage()` called `foldchanges` that has `names()` attribute set to ENTREZIDs

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

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

```
attributes(keggres)
```

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

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

```
                           p.geomean stat.mean        p.val       q.val
hsa04110 Cell cycle      8.995727e-06 -4.378644 8.995727e-06 0.001889103
hsa03030 DNA replication 9.424076e-05 -3.951803 9.424076e-05 0.009841047
                         set.size         exp1
hsa04110 Cell cycle           121 8.995727e-06
hsa03030 DNA replication       36 9.424076e-05
```

```
pathview(gene.data=foldchanges, pathway.id="hsa04110")
```

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

```
Info: Working in directory /Users/hanheejo/Desktop/1. UCSD/1. Class/7. Winter 2025/BGGN213/Cl
```

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

```
pathview(gene.data=foldchanges, pathway.id="hsa03030")
```

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

```
Info: Working in directory /Users/hanheejo/Desktop/1. UCSD/1. Class/7. Winter 2025/BGGN213/Cl
```

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

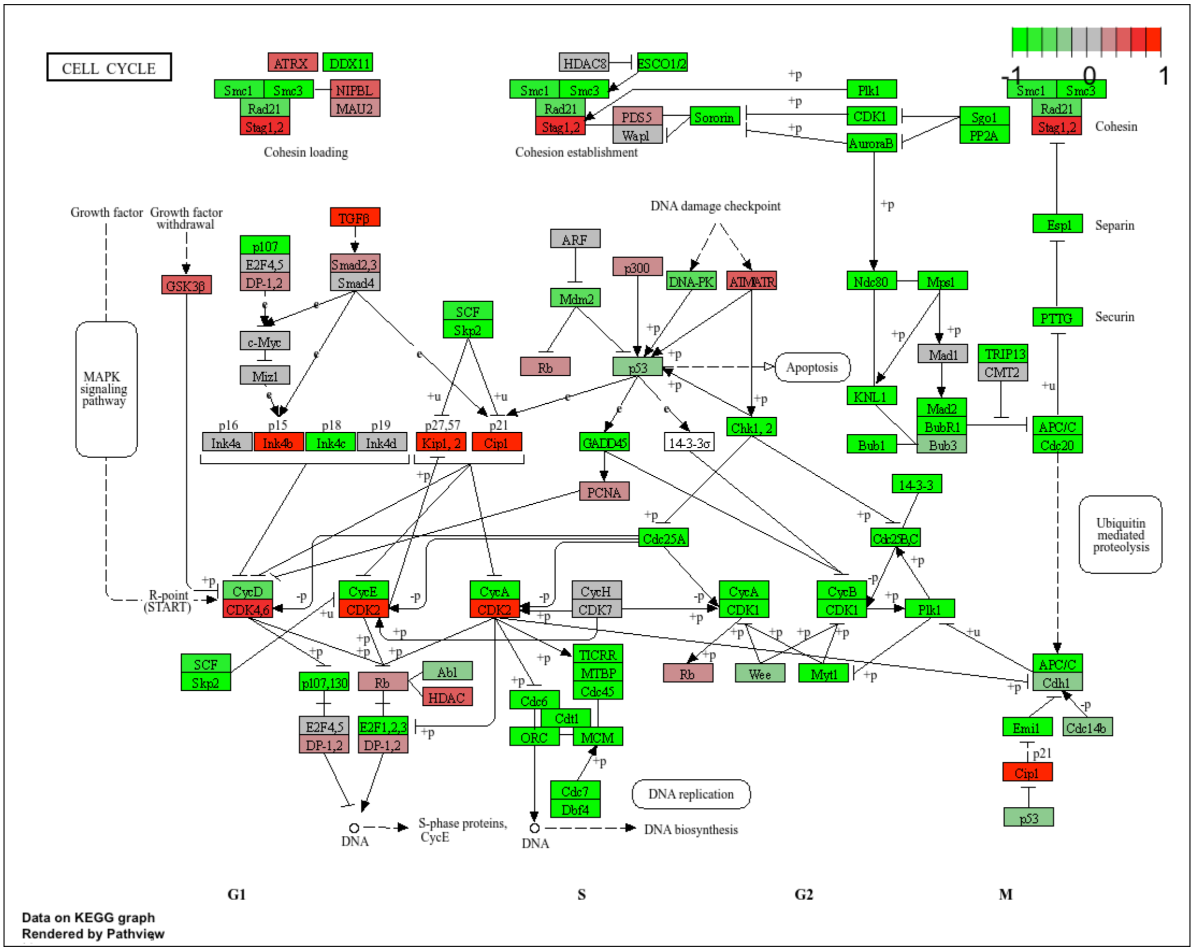
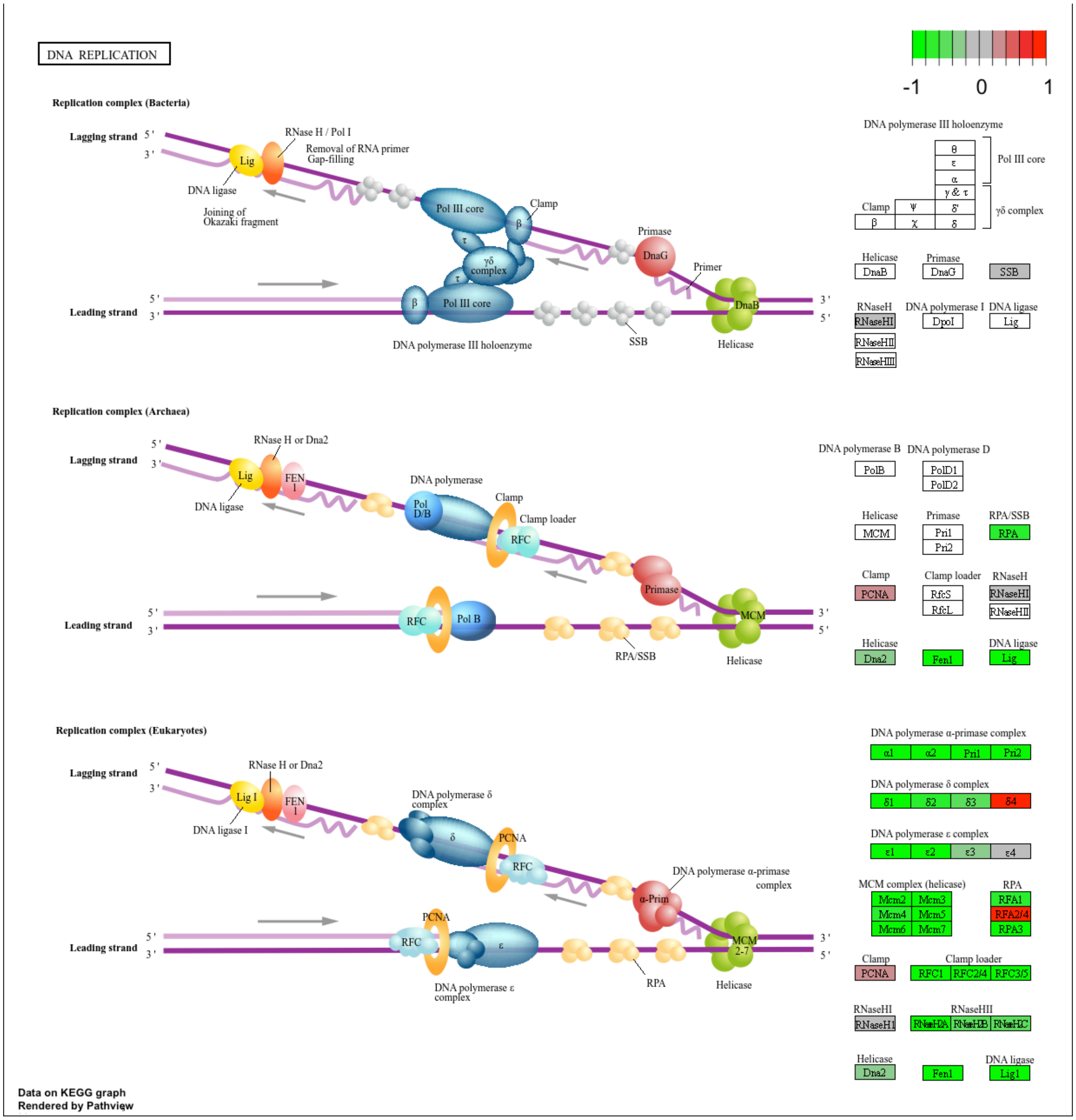


Figure 1: Cell cycle is affected


DNA REPLICATION
-1
0
1
Replication complex (Bacteria)
Lagging strand
Leading strand
5'
3'
Lig
DNA ligase
Joining of Okazaki fragment
RNase H / Pol I
Removal of RNA primer
Gap-filling
Pol III core
Clamp
β
τ
γδ complex
Primase
DnaG
Primer
DnaB
Helicase
SSB
DNA polymerase III holoenzyme
DNA polymerase III holoenzyme
θ
ε
α
γ & τ
ψ
δ'
χ
δ
Pol III core
γδ complex
Helicase
DnaB
Primase
DnaG
SSB
RNaseH
RNaseHI
RNaseHII
RNaseHIII
DNA polymerase I
DpoI
DNA ligase
Lig
Replication complex (Archaea)
RNase H or Dna2
FEN 1
DNA polymerase
Pol D/B
Clamp loader
RFC
Primase
MCM
Pol B
RPA/SSB
DNA polymerase B
PolB
DNA polymerase D
PolD1
PolD2
Helicase
MCM
Primase
Pri1
Pri2
RPA/SSB
RPA
Clamp
PCNA
Clamp loader
RfcS
RfcL
RNaseH
RNaseHI
RNaseHII
Helicase
Dna2
Fen1
DNA ligase
Lig
Replication complex (Eukaryotes)
RNase H or Dna2
Lig I
DNA ligase I
FEN 1
DNA polymerase δ complex
δ
PCNA
RFC
DNA polymerase α-primase complex
α-Prim
MCM 2-7
ε
DNA polymerase ε complex
RPA
Helicase
DNA polymerase α-primase complex
α1
α2
Pri1
Pri2
DNA polymerase δ complex
δ1
δ2
δ3
δ4
DNA polymerase ε complex
ε1
ε2
ε3
ε4
MCM complex (helicase)
Mcm2
Mcm3
Mcm4
Mcm5
Mcm6
Mcm7
RPA
RPA1
RPA2/4
RPA3
Clamp
PCNA
Clamp loader
RFC1
RFC2/4
RFC3/5
RNaseHI
RNaseH1
RNaseHII
RNaseH2A
RNaseH2B
RNaseH2C
Helicase
Dna2
Fen1
DNA ligase
Lig1
Data on KEGG graph
Rendered by Pathview

## Gene Ontology (GO)

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

# Focus just on GO Biological Process (BP)

gobpsets = go.sets.hs[go.subs.hs$BP]

gobpres = gage(foldchanges, gsets=gobpsets)
```

```
head(gobpres$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             5.843127e-12      376 1.536227e-15
GO:0000280 nuclear division              5.843127e-12      352 4.286961e-15
GO:0007067 mitosis                       5.843127e-12      352 4.286961e-15
GO:0000087 M phase of mitotic cell cycle 1.195965e-11      362 1.169934e-14
GO:0007059 chromosome segregation        1.659009e-08      142 2.028624e-11
GO:0000236 mitotic prometaphase          1.178690e-07       84 1.729553e-10
```

## Reactome

We can use reatome via R or via their fancy new website interface. The web interface wants a set of ENTREZ ID values for your genes of interest. Let's generate that.

```
inds <- abs(res$log2FoldChange) >= 2 & res$padj <=0.05
top.genes <- res$entrez[inds]
```

```
write.table(top.genes, file="top_genes.txt", row.names=FALSE, col.names=FALSE, quote=FALSE
```