Class 14: RNASeq mini project

Nathan Phan (A17395036)

Table of contents

Background

Here we work through a complete RNASeq analysis project. The input data comes from a knock-down experiment of a HOX gene.

Data Import

Reading the `counts` and `metadata` CSV files

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

Check on data structure

```
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
```

```
ncol(counts)
```

```
[1] 7
```

```
nrow(metadata)
```

```
[1] 6
```

looks like we need to get rid of the first "length" colunn of our `counts` object.

```
cleancounts <-counts[ , -1]
```

```
colnames(cleancounts)
```

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

```
metadata$id
```

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

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

```
[1] TRUE
```

Remove zero count genes

There are lots of genes with zerp counts. We can remove these from further analysis.

```r
head(cleancounts)
```

```
                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
```

```r
to.keep.inds <- rowSums(cleancounts) > 0
nonzero_counts <- cleancounts[to.keep.inds,]
```

DESeq analysis

Load the package

```r
library(DESeq2)
```

```
Warning: package 'matrixStats' was built under R version 4.5.2
```

Setup DESeq object

```r
dds <- DESeqDataSetFromMatrix(countData = nonzero_counts,
                              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
```

get 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 1.43990e-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 0.5215598   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
```

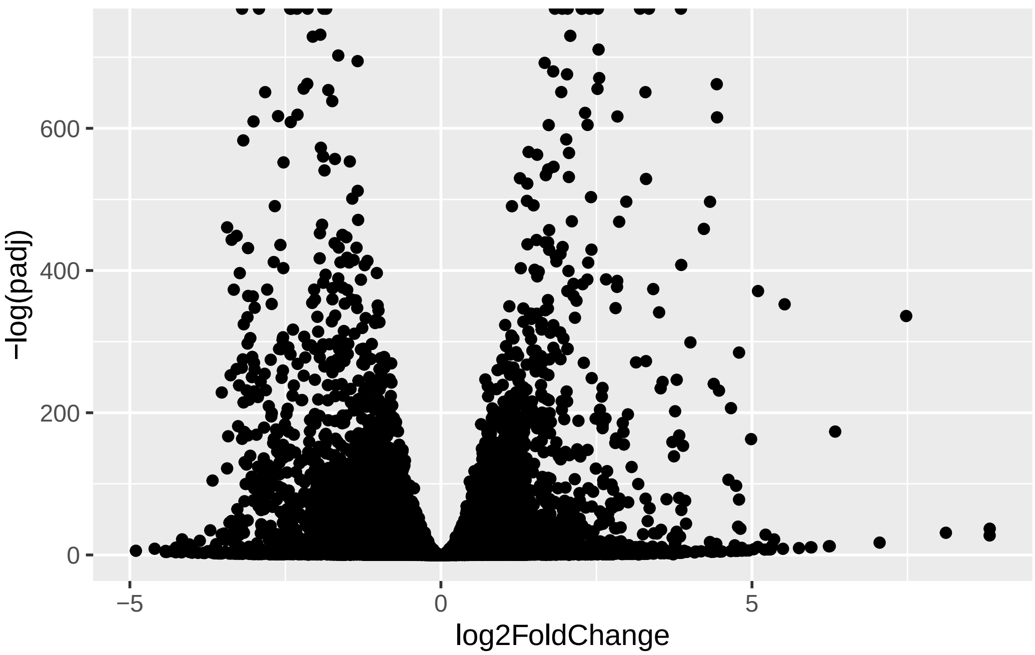
Data visualization

Volcano plot

```r
library(ggplot2)

ggplot(res) +
  aes(log2FoldChange, -log(padj)) +
  geom_point()
```

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

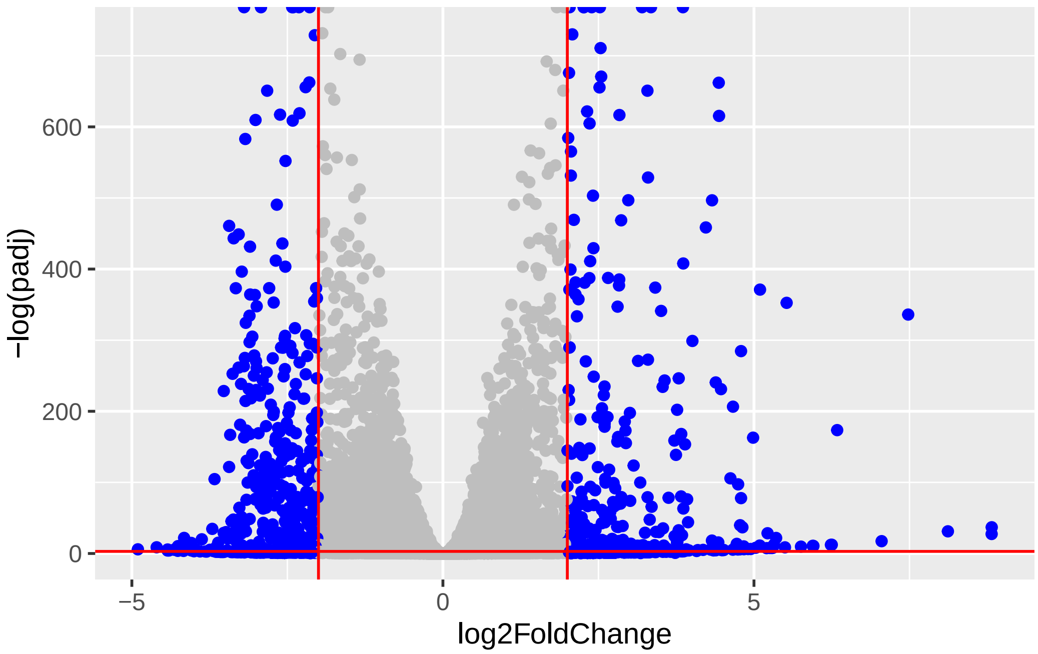


Add threshold lines for fold-change and P-value and color our subset of genes that make these threshold cut-offs in the plot

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

ggplot(res) +
 aes(log2FoldChange, -log(padj)) +
  geom_point(col=mycols) +
  geom_vline(xintercept = c(-2,2), col="red") +
  geom_hline(yintercept = -log(0.05), col="red")
```

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



Add Annotation

Add gene symbols and entrez ids

```
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"     
```

```
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
```

```
head(res, 10)
```

```
log2 fold change (MLE): condition hoxa1 kd vs control sirna
Wald test p-value: condition hoxa1 kd vs control sirna
DataFrame with 10 rows and 9 columns
                   baseMean log2FoldChange     lfcSE       stat      pvalue
                  <numeric>      <numeric> <numeric>  <numeric>   <numeric>
ENSG00000279457   29.913579      0.1792571 0.3248216   0.551863 5.81042e-01
ENSG00000187634  183.229650      0.4264571 0.1402658   3.040350 2.36304e-03
ENSG00000188976 1651.188076     -0.6927205 0.0548465 -12.630158 1.43990e-36
ENSG00000187961  209.637938      0.7297556 0.1318599   5.534326 3.12428e-08
ENSG00000187583   47.255123      0.0405765 0.2718928   0.149237 8.81366e-01
ENSG00000187642   11.979750      0.5428105 0.5215598   1.040744 2.97994e-01
ENSG00000188290  108.922128      2.0570638 0.1969053  10.446970 1.51282e-25
ENSG00000187608  350.716868      0.2573837 0.1027266   2.505522 1.22271e-02
```

```
ENSG00000188157 9128.439422      0.3899088 0.0467163   8.346304 7.04321e-17
ENSG00000237330    0.158192      0.7859552 4.0804729   0.192614 8.47261e-01
                       padj      Symbol      Entrez                   name
                  <numeric> <character> <character>            <character>
ENSG00000279457 6.86555e-01          NA          NA                     NA
ENSG00000187634 5.15718e-03      SAMD11      148398 sterile alpha motif ..
ENSG00000188976 1.76549e-35       NOC2L       26155 NOC2 like nucleolar ..
ENSG00000187961 1.13413e-07      KLHL17      339451 kelch like family me..
ENSG00000187583 9.19031e-01     PLEKHN1       84069 pleckstrin homology ..
ENSG00000187642 4.03379e-01       PERM1       84808 PPARGC1 and ESRR ind..
ENSG00000188290 1.30538e-24        HES4       57801 hes family bHLH tran..
ENSG00000187608 2.37452e-02       ISG15        9636 ISG15 ubiquitin like..
ENSG00000188157 4.21963e-16        AGRN      375790                  agrin
ENSG00000237330          NA      RNF223      401934 ring finger protein ..
```

Pathway Analysis

Run Gage analysis with KEGG

```r
library(gage)
```

```r
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).
##############################################################################
```

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

```
kegg.sets.hs = kegg.sets.hs[sigmet.idx.hs]
```

We need a named vector

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

```
[1]  0.17925708  0.42645712 -0.69272046  0.72975561  0.04057653  0.54281049
```

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

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

# Focus on signaling and metabolic pathways only
kegg.sets.hs = kegg.sets.hs[sigmet.idx.hs]

# Examine the first 3 pathways
head(kegg.sets.hs, 3)
```

```
$`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"

$`hsa00230 Purine metabolism`
  [1] "100"    "10201"  "10606"  "10621"  "10622"  "10623"  "107"    "10714"
  [9] "108"    "10846"  "109"    "111"    "11128"  "11164"  "112"    "113"
 [17] "114"    "115"    "122481" "122622" "124583" "132"    "158"    "159"
 [25] "1633"   "171568" "1716"   "196883" "203"    "204"    "205"    "221823"
 [33] "2272"   "22978"  "23649"  "246721" "25885"  "2618"   "26289"  "270"
 [41] "271"    "27115"  "272"    "2766"   "2977"   "2982"   "2983"   "2984"  
 [49] "2986"   "2987"   "29922"  "3000"   "30833"  "30834"  "318"    "3251"  
 [57] "353"    "3614"   "3615"   "3704"   "377841" "471"    "4830"   "4831"  
 [65] "4832"   "4833"   "4860"   "4881"   "4882"   "4907"   "50484"  "50940" 
 [73] "51082"  "51251"  "51292"  "5136"   "5137"   "5138"   "5139"   "5140"  
 [81] "5141"   "5142"   "5143"   "5144"   "5145"   "5146"   "5147"   "5148"  
 [89] "5149"   "5150"   "5151"   "5152"   "5153"   "5158"   "5167"   "5169"  
 [97] "51728"  "5198"   "5236"   "5313"   "5315"   "53343"  "54107"  "5422"  
[105] "5424"   "5425"   "5426"   "5427"   "5430"   "5431"   "5432"   "5433"  
[113] "5434"   "5435"   "5436"   "5437"   "5438"   "5439"   "5440"   "5441"  
[121] "5471"   "548644" "55276"  "5557"   "5558"   "55703"  "55811"  "55821" 
[129] "5631"   "5634"   "56655"  "56953"  "56985"  "57804"  "58497"  "6240"  
[137] "6241"   "64425"  "646625" "654364" "661"    "7498"   "8382"   "84172" 
[145] "84265"  "84284"  "84618"  "8622"   "8654"   "87178"  "8833"   "9060"  
[153] "9061"   "93034"  "953"    "9533"   "954"    "955"    "956"    "957"   
[161] "9583"   "9615"  
```

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

```
                                         p.geomean stat.mean p.val q.val
hsa00232 Caffeine metabolism                    NA       NaN    NA    NA
hsa00983 Drug metabolism - other enzymes        NA       NaN    NA    NA
                                         set.size exp1
hsa00232 Caffeine metabolism                    0   NA
hsa00983 Drug metabolism - other enzymes        0   NA
```

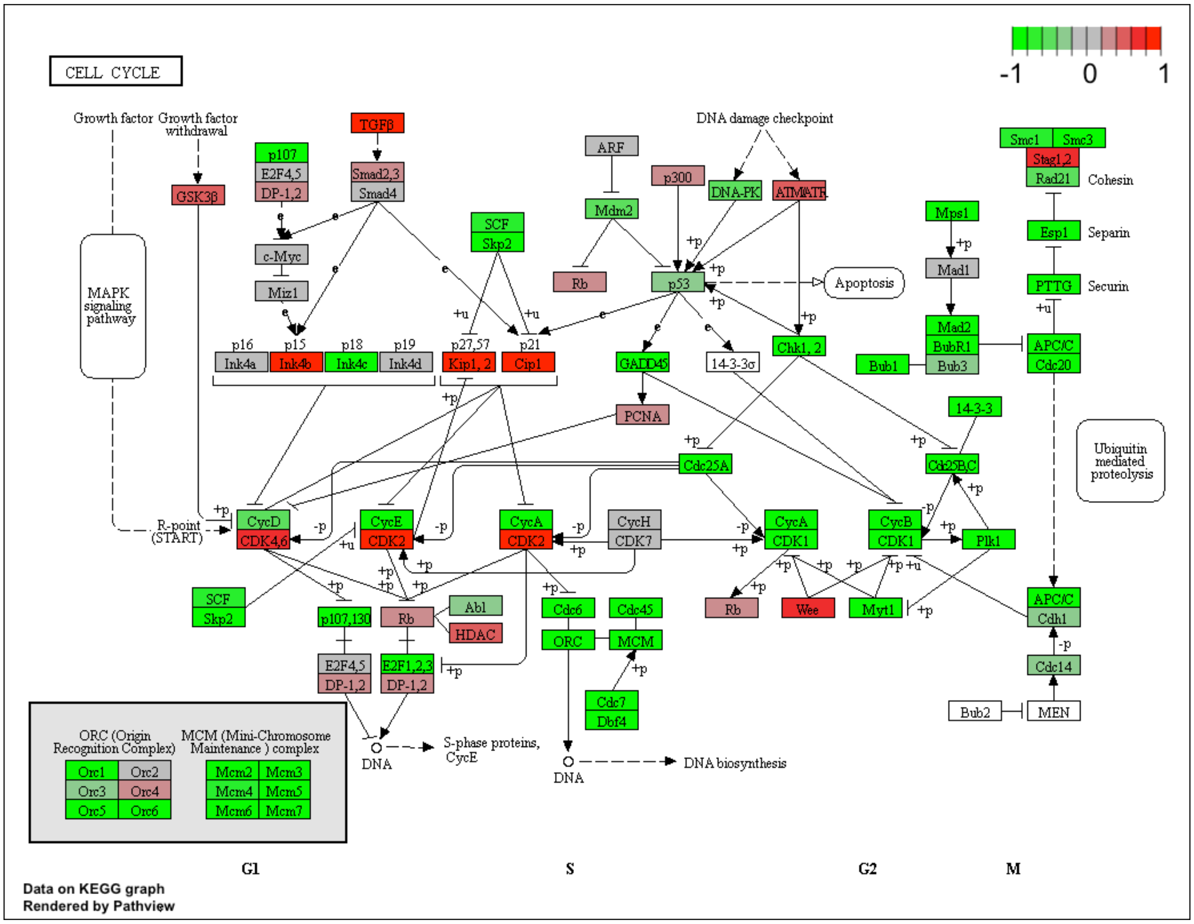
```r
pathview(gene.data=foldchanges, pathway.id="hsa04110")
```

```
Warning: None of the genes or compounds mapped to the pathway!
Argument gene.idtype or cpd.idtype may be wrong.
```

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

```
Info: Working in directory C:/Users/Kawai/OneDrive/Desktop/BIMM 143/Class 14
```

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



Data on KEGG graph
Rendered by Pathview

GO terms

Same analysis but using GO genesets rather than KEGG

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

# Focus on Biological Process subset of GO
gobpsets = go.sets.hs[go.subs.hs$BP]

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

lapply(gobpres, head)
```

```
$greater
```

```
                                               p.geomean stat.mean p.val q.val
GO:0000002 mitochondrial genome maintenance           NA       NaN    NA    NA
GO:0000003 reproduction                               NA       NaN    NA    NA
GO:0000012 single strand break repair                 NA       NaN    NA    NA
GO:0000018 regulation of DNA recombination            NA       NaN    NA    NA
GO:0000019 regulation of mitotic recombination        NA       NaN    NA    NA
GO:0000022 mitotic spindle elongation                 NA       NaN    NA    NA
                                               set.size exp1
GO:0000002 mitochondrial genome maintenance           0   NA
GO:0000003 reproduction                               0   NA
GO:0000012 single strand break repair                 0   NA
GO:0000018 regulation of DNA recombination            0   NA
GO:0000019 regulation of mitotic recombination        0   NA
GO:0000022 mitotic spindle elongation                 0   NA

$less
                                               p.geomean stat.mean p.val q.val
GO:0000002 mitochondrial genome maintenance           NA       NaN    NA    NA
GO:0000003 reproduction                               NA       NaN    NA    NA
GO:0000012 single strand break repair                 NA       NaN    NA    NA
GO:0000018 regulation of DNA recombination            NA       NaN    NA    NA
GO:0000019 regulation of mitotic recombination        NA       NaN    NA    NA
GO:0000022 mitotic spindle elongation                 NA       NaN    NA    NA
                                               set.size exp1
GO:0000002 mitochondrial genome maintenance           0   NA
GO:0000003 reproduction                               0   NA
GO:0000012 single strand break repair                 0   NA
GO:0000018 regulation of DNA recombination            0   NA
GO:0000019 regulation of mitotic recombination        0   NA
GO:0000022 mitotic spindle elongation                 0   NA

$stats
                                               stat.mean exp1
GO:0000002 mitochondrial genome maintenance          NaN   NA
GO:0000003 reproduction                              NaN   NA
GO:0000012 single strand break repair                NaN   NA
GO:0000018 regulation of DNA recombination           NaN   NA
GO:0000019 regulation of mitotic recombination       NaN   NA
GO:0000022 mitotic spindle elongation                NaN   NA
```

Reactome

Lots of folks like the reactome web interface. You can also run this as an R function but lets look at the website first. < https://reactome.org/

The website wants a text file with one gene symbol per line of the genes you want to map to pathways.

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

```
NULL
```

```
#res$symbols
print(paste("Total number of significant genes:", length(sig_genes)))
```

```
[1] "Total number of significant genes: 0"
```

and write out a file

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

> Q: What pathway has the most significant "Entities p-value"? Do the most significant pathways listed match your previous KEGG results? What factors could cause differences between the two methods?

The pathway with the lowest Entities p-value is the most significant. The Reactome results don't exactly match the KEGG pathways, because the two methods use different pathway databases, different gene sets, and different statistical approaches, which naturally leads to differences in which pathways appear most enriched.

Save Our Results

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