class 13: RNA-Seq analysis mini-project

Section 1. Differential Expression Analysis

We need to load our data - metadata and countdata

```
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

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
```

```
metaFile <- "https://bioboot.github.io/bimm143_W18/class-material/GSE37704_metadata.csv"
countFile <- "https://bioboot.github.io/bimm143_W18/class-material/GSE37704_featurecounts.

# Import metadata and take a peak
colData = read.csv(metaFile, row.names=1)
head(colData)
```

```
            condition
SRR493366 control_sirna
SRR493367 control_sirna
SRR493368 control_sirna
SRR493369      hoxa1_kd
SRR493370      hoxa1_kd
SRR493371      hoxa1_kd
```

```
# Import countdata
countData = read.csv(countFile, row.names=1)
head(countData)
```

```
                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
```

We need the countData and colData files to match up so we will need to remove that odd first column in countData namely contData$length.

> Q. Complete the code below to remove the troublesome first column from countData.

```
# Note we need to remove the odd first $length col
countData <- as.matrix(countData[,-1])
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
```

There are too many zeros so we should remove them.

> 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).

```
# Filter count data where you have 0 read count across all samples.
countData = countData[rowSums(countData)>1, ]
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

Now lets setup the DESeqDataSet object required for the DESeq() function and then run the DESeq pipeline.

```
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
```

```
dds = DESeq(dds)
```

```
estimating size factors
```

```
estimating dispersions
```

```
gene-wise dispersion estimates
```

```
mean-dispersion relationship
```

```
final dispersion estimates
```

```
fitting model and testing
```

```
dds
```

```
class: DESeqDataSet
dim: 15280 6
metadata(1): version
assays(4): counts mu H cooks
rownames(15280): ENSG00000279457 ENSG00000187634 ... ENSG00000276345
```

```
  ENSG00000271254
rowData names(22): baseMean baseVar ... deviance maxCooks
colnames(6): SRR493366 SRR493367 ... SRR493370 SRR493371
colData names(2): condition sizeFactor
```

Now let's get the results for the HoxA1 knockdown versus control siRNA.

```
res = results(dds, contrast=c("condition", "hoxa1_kd", "control_sirna"))
```

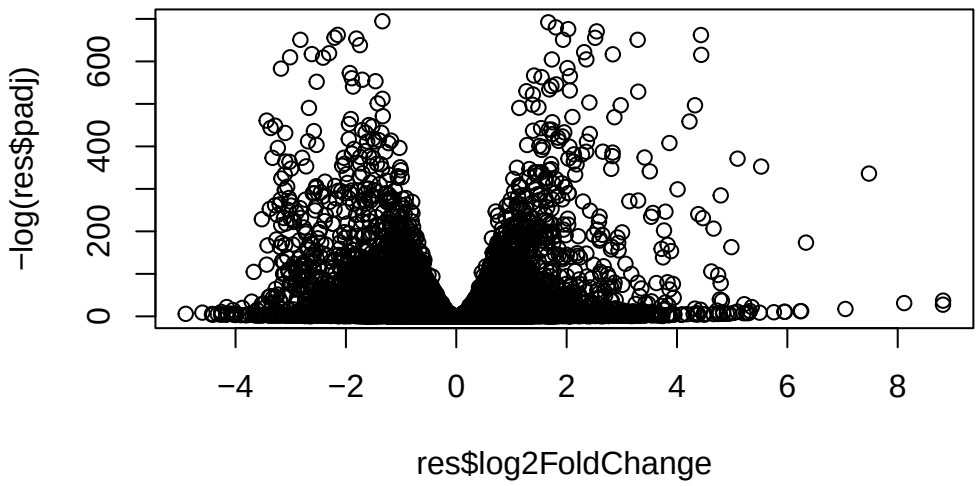
Q. Call the summary() function on your results to get a sense of how many genes are up or down-regulated at the default 0.1 p-value cutoff.

```
summary(res)
```

```
out of 15280 with nonzero total read count
adjusted p-value < 0.1
LFC > 0 (up)       : 4351, 28%
LFC < 0 (down)     : 4399, 29%
outliers [1]       : 0, 0%
low counts [2]     : 590, 3.9%
(mean count < 1)
[1] see 'cooksCutoff' argument of ?results
[2] see 'independentFiltering' argument of ?results
```

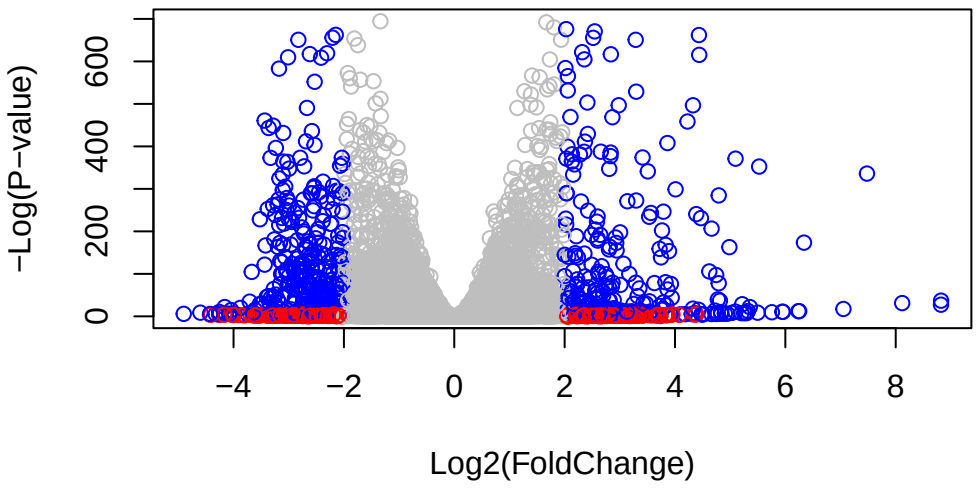
Let's make a volcano plot!

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



Q. Improve this plot by completing the below code, which adds color and axis labels

```
# Make a color vector for all genes
mycols <- rep("gray", nrow(res) )

# Color red the genes with absolute fold change above 2
mycols[ abs(res$log2FoldChange) > 2 ] <- "red"

# Color blue those with adjusted p-value less than 0.01
#  and absolute fold change more than 2
inds <- (res$padj<0.01) & (abs(res$log2FoldChange) > 2 )
mycols[ inds ] <- "blue"

plot( res$log2FoldChange, -log(res$padj), col=mycols, xlab="Log2(FoldChange)", ylab="-Log(
```



Let's add gene annotation.

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

```
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   29.9136      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
```

```
ENSG00000188290  108.9221      2.0570638 0.1969053  10.446970 1.51282e-25
ENSG00000187608  350.7169      0.2573837 0.1027266   2.505522 1.22271e-02
ENSG00000188157 9128.4394      0.3899088 0.0467163   8.346304 7.04321e-17
ENSG00000131591  156.4791      0.1965923 0.1456109   1.350121 1.76977e-01
                       padj      symbol      entrez                   name
                  <numeric> <character> <character>            <character>
ENSG00000279457 6.85033e-01          NA          NA                     NA
ENSG00000187634 5.14039e-03      SAMD11      148398 sterile alpha motif ..
ENSG00000188976 1.75974e-35       NOC2L       26155 NOC2 like nucleolar ..
ENSG00000187961 1.13044e-07      KLHL17      339451 kelch like family me..
ENSG00000187583 9.19159e-01     PLEKHN1       84069 pleckstrin homology ..
ENSG00000187642 4.02066e-01       PERM1       84808 PPARGC1 and ESRR ind..
ENSG00000188290 1.30113e-24        HES4       57801 hes family bHLH tran..
ENSG00000187608 2.36679e-02       ISG15        9636 ISG15 ubiquitin like..
ENSG00000188157 4.20589e-16        AGRN      375790                  agrin
ENSG00000131591 2.60893e-01    C1orf159       54991 chromosome 1 open re..
```

Q.Finally for this section let's reorder these results by adjusted p-value and save them to a CSV file in your current project directory.

```
res = res[order(res$pvalue),]
write.csv(res, file="deseq_results.csv")
```

Section 2. Pathway Analysis

First we need to install these required bioconductor packages.

Now we can load the packages and setup the KEGG data-sets we need.

```
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(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"
```

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

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

```
     1266     54855      1465     51232      2034      2317
-2.422719  3.201955 -2.313738 -2.059631 -1.888019 -1.649792
```

Now, let's run the gage pathway analysis.

```
# Get the results
keggres = gage(foldchanges, gsets=kegg.sets.hs)
```

Now lets look at the object returned from gage().

```
attributes(keggres)
```

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

```
# Look at the first few down (less) pathways
head(keggres$less)
```

```
                                      p.geomean stat.mean        p.val
hsa04110 Cell cycle                1.003993e-05 -4.353454 1.003993e-05
hsa03030 DNA replication           8.909558e-05 -3.968611 8.909558e-05
hsa03013 RNA transport             1.470985e-03 -3.007794 1.470985e-03
```

```
hsa04114 Oocyte meiosis               1.946905e-03 -2.921710 1.946905e-03
hsa03440 Homologous recombination     2.941989e-03 -2.868141 2.941989e-03
hsa00010 Glycolysis / Gluconeogenesis 6.059196e-03 -2.558327 6.059196e-03
                                            q.val set.size         exp1
hsa04110 Cell cycle                   0.001606390      120 1.003993e-05
hsa03030 DNA replication              0.007127646       36 8.909558e-05
hsa03013 RNA transport                0.077876201      143 1.470985e-03
hsa04114 Oocyte meiosis               0.077876201       99 1.946905e-03
hsa03440 Homologous recombination     0.094143663       28 2.941989e-03
hsa00010 Glycolysis / Gluconeogenesis 0.161578551       48 6.059196e-03
```

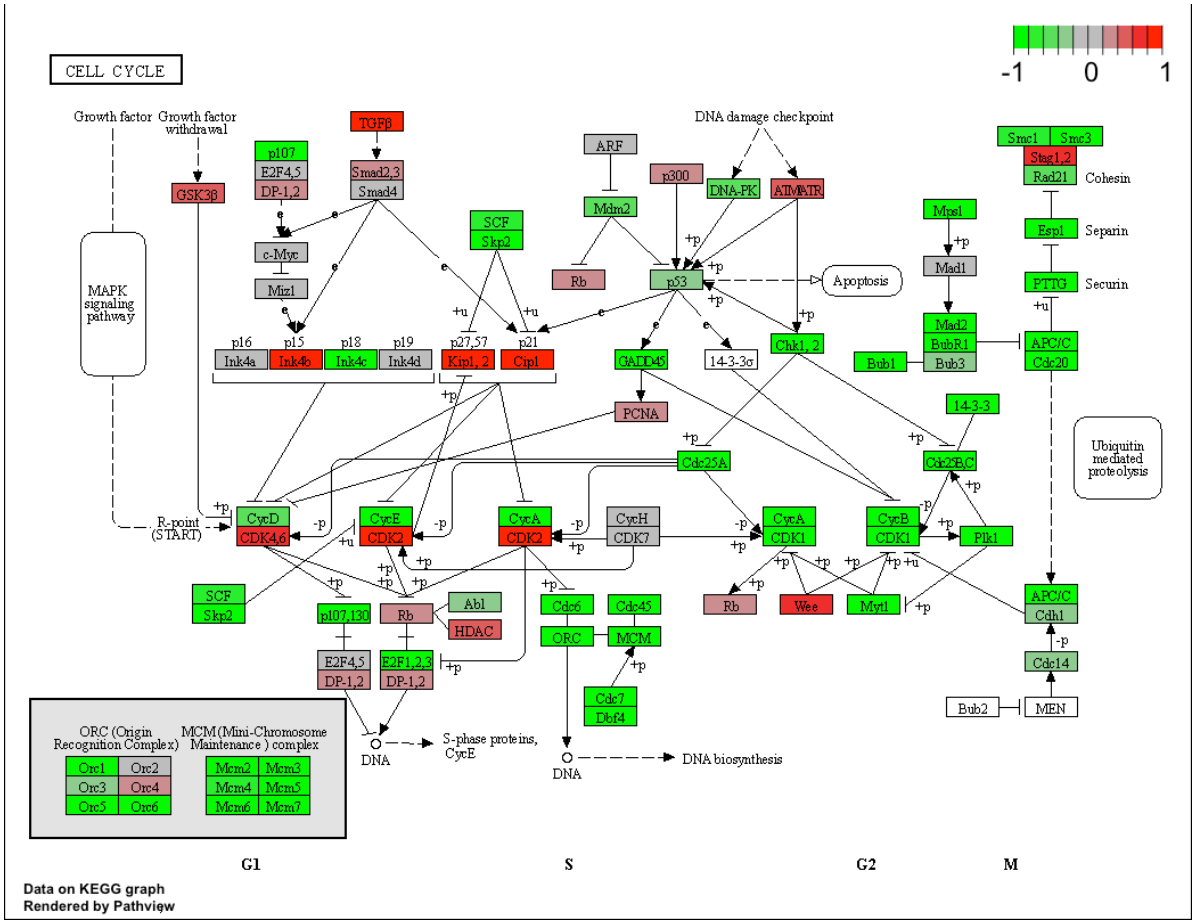
Now, let's try out the pathview() function from the pathview package to make a pathway plot with our RNA-Seq expression results shown in color.

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

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

```
Info: Working in directory /Users/jenniferquach/Desktop/Bimm 143/class 13
```

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



Now, let's process our results a bit more to automagicaly pull out the top 5 upregulated pathways.

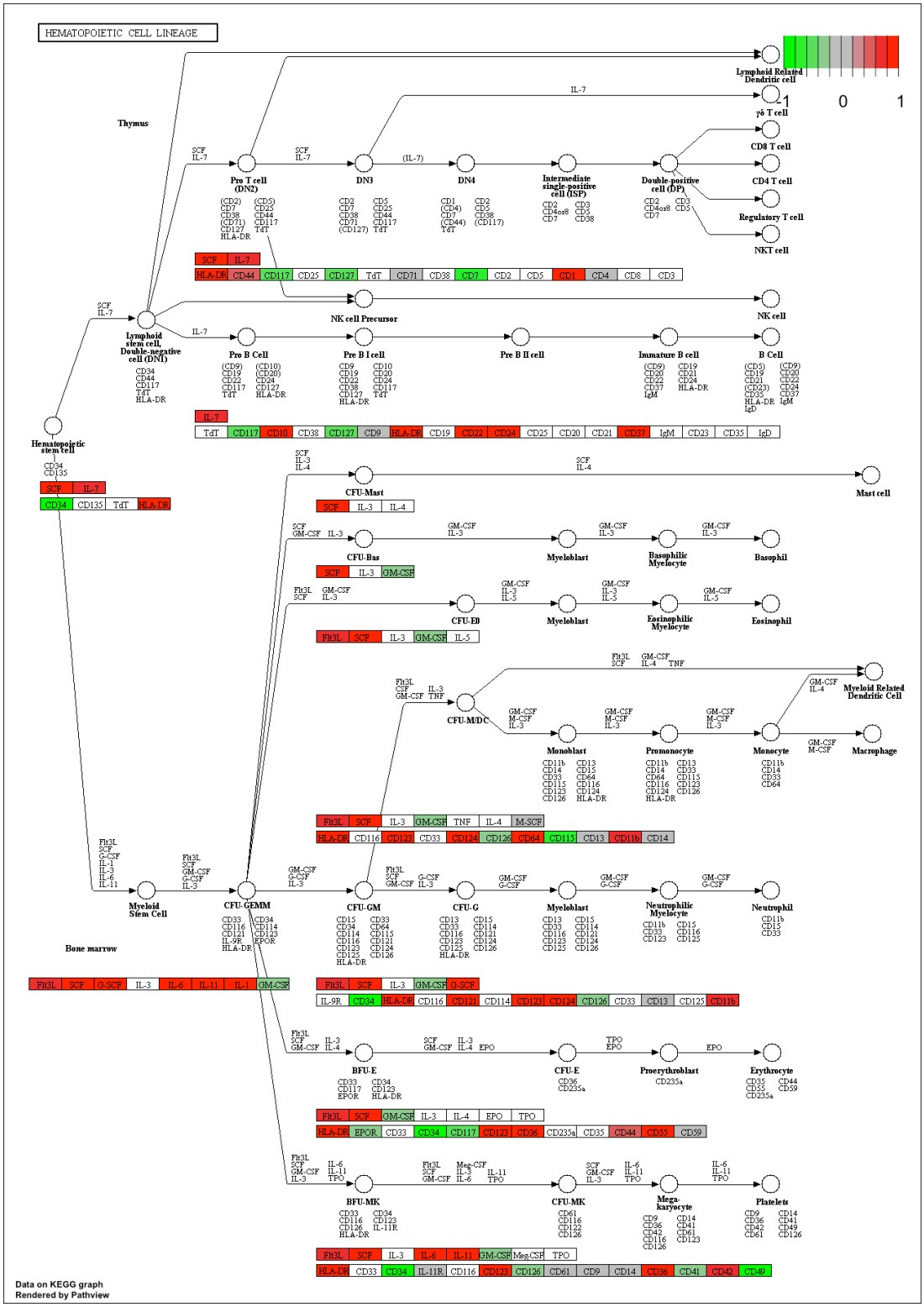
```
## Focus on top 5 upregulated pathways here for demo purposes only
keggrespathways <- rownames(keggres$greater)[1:5]

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

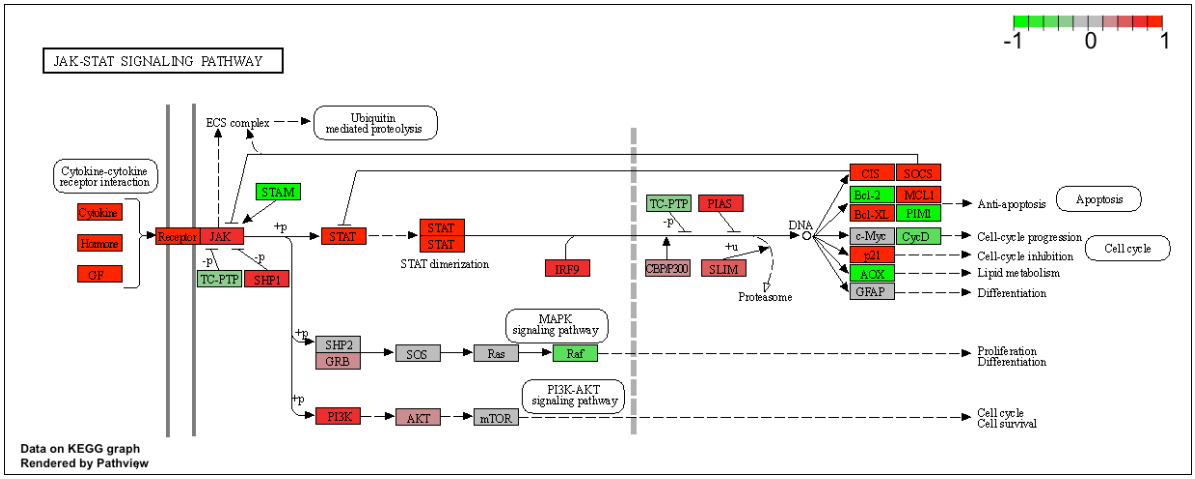
```
[1] "hsa04640" "hsa04630" "hsa04142" "hsa00140" "hsa04740"
```

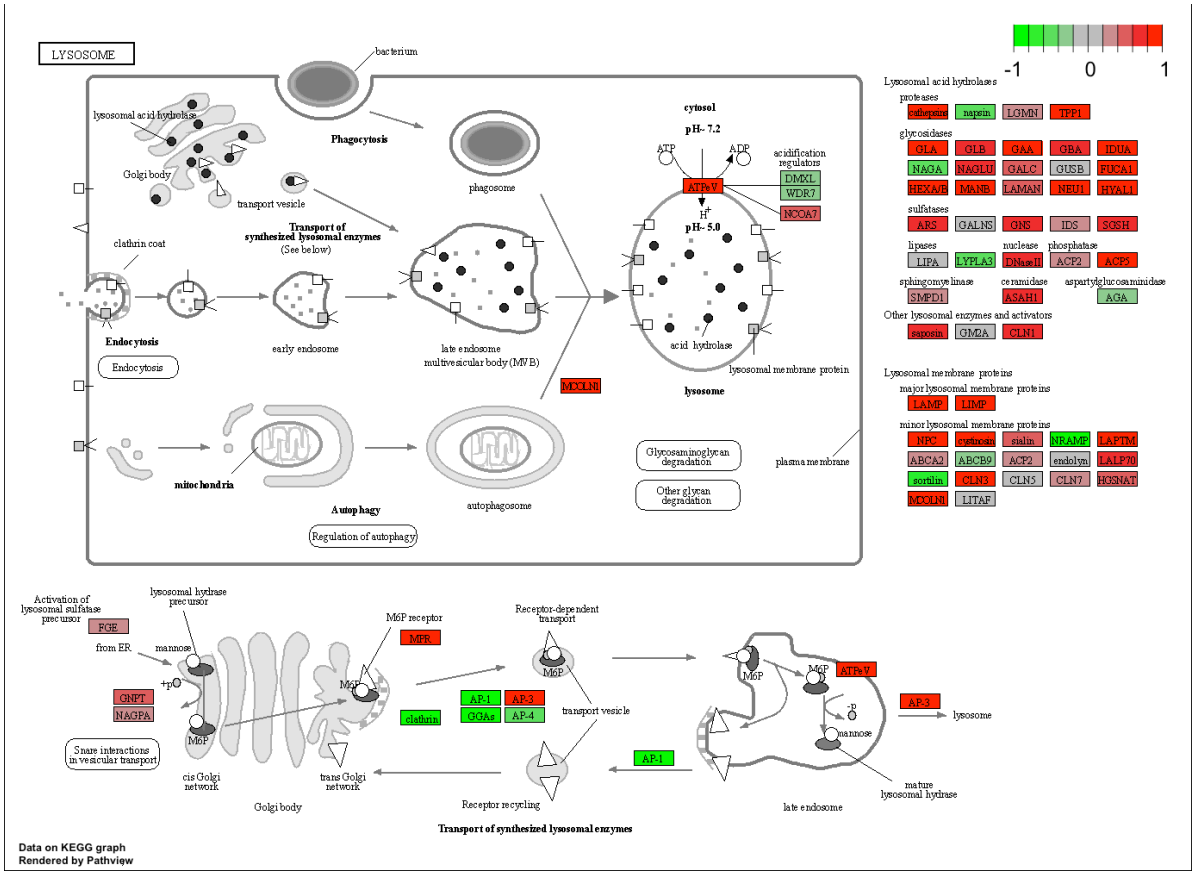
Finally, lets pass these IDs in keggresids to the pathview() function to draw plots for all the top 5 pathways.

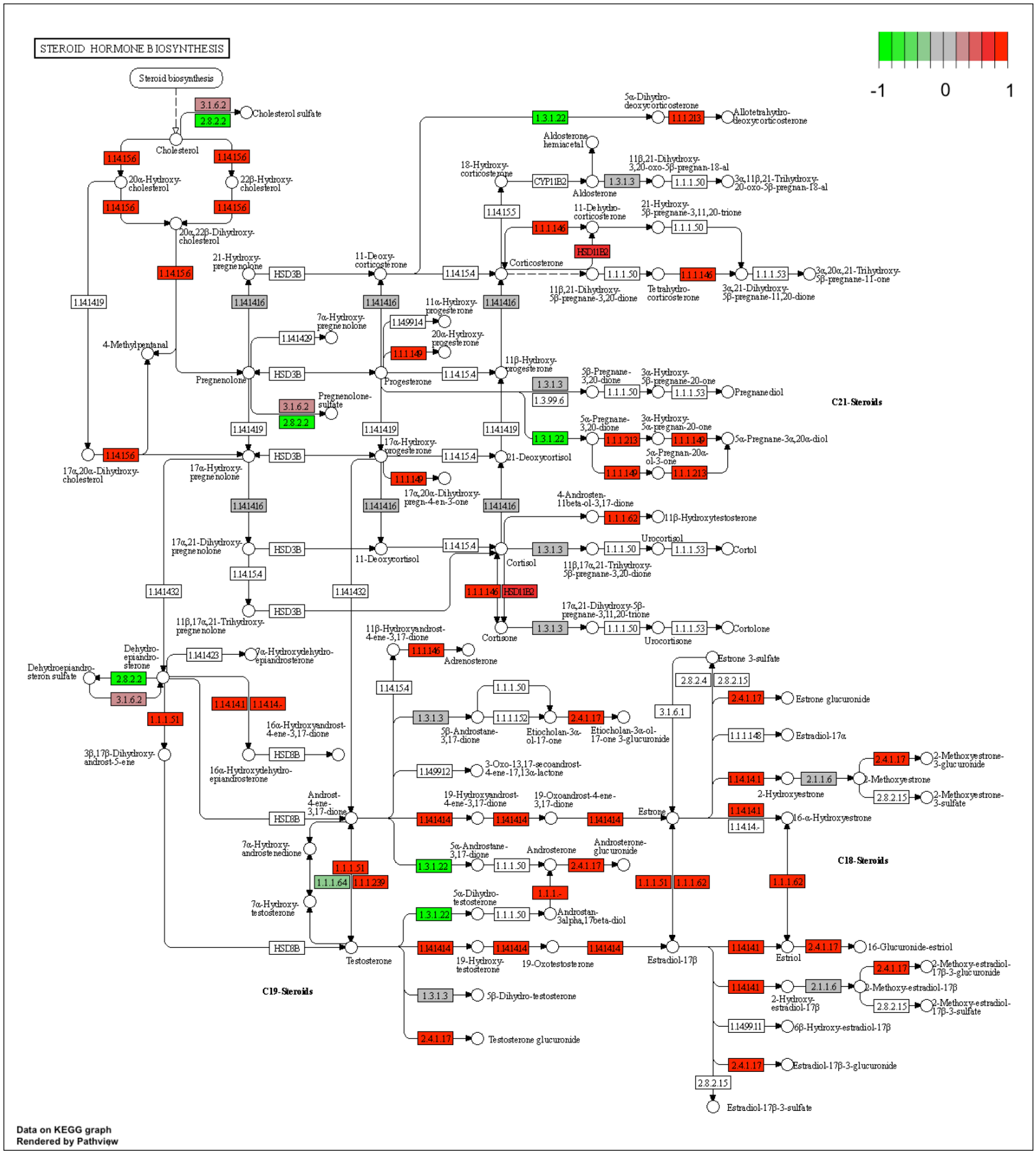
```
pathview(gene.data=foldchanges, pathway.id=keggresids, species="hsa")
```

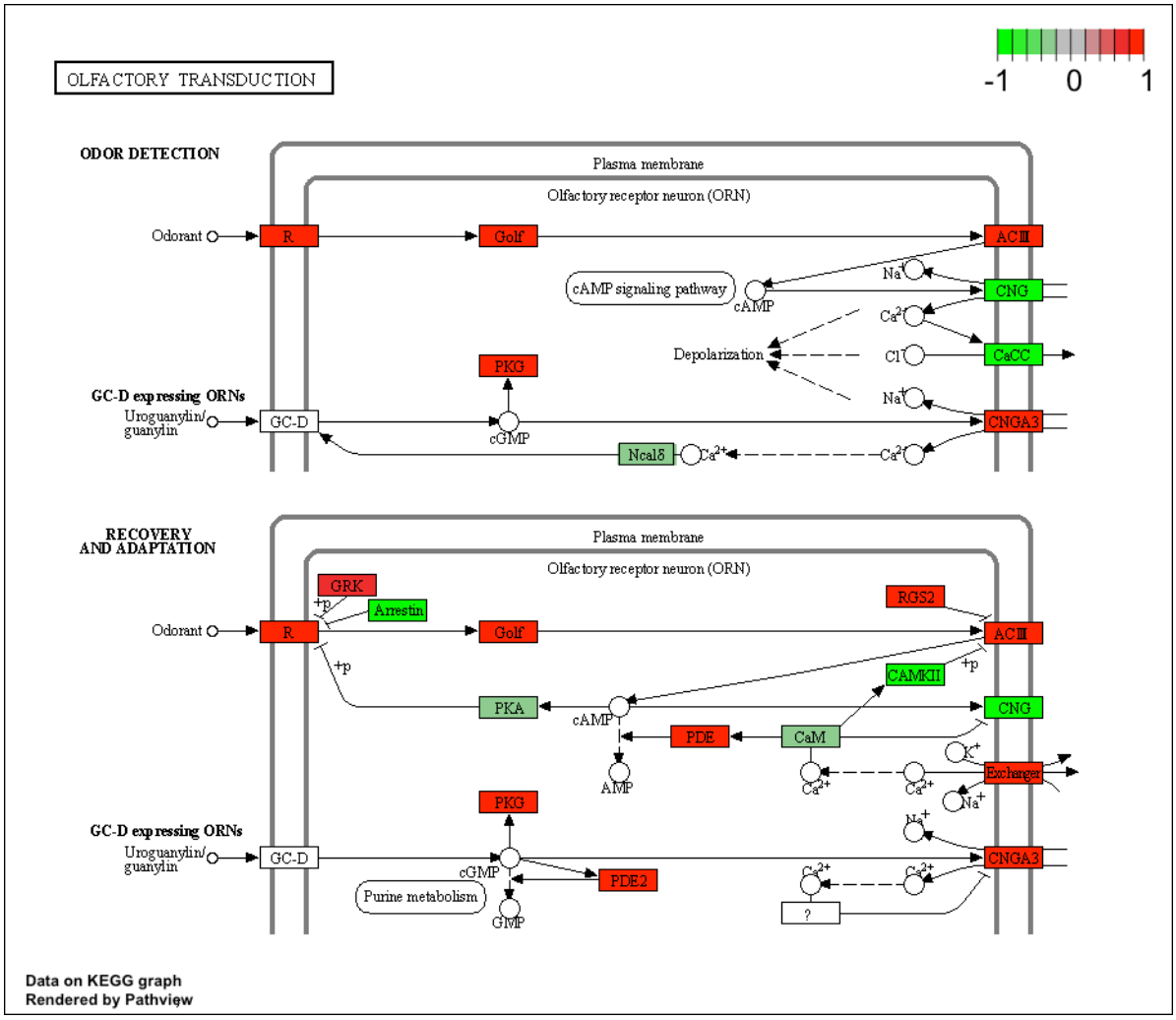
```
'select()' returned 1:1 mapping between keys and columns

Info: Working in directory /Users/jenniferquach/Desktop/Bimm 143/class 13

Info: Writing image file hsa04640.pathview.png

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

Info: Working in directory /Users/jenniferquach/Desktop/Bimm 143/class 13

Info: Writing image file hsa04630.pathview.png

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

Info: Working in directory /Users/jenniferquach/Desktop/Bimm 143/class 13

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 /Users/jenniferquach/Desktop/Bimm 143/class 13

Info: Writing image file hsa00140.pathview.png

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

Info: Working in directory /Users/jenniferquach/Desktop/Bimm 143/class 13

Info: Writing image file hsa04740.pathview.png

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


HEMATOPOIETIC CELL LINEAGE
Lymphoid Related Dendritic cell
γδ T cell
CD8 T cell
CD4 T cell
Regulatory T cell
NKT cell
NK cell
Thymus
Pre T cell (DN2)
DN3
DN4
Intermediate single-positive cell (ISP)
Double-positive cell (DP)
NK cell Precursor
Lymphoid stem cell, Double-negative cell (DN1)
Pro B Cell
Pre B I cell
Pre B II cell
Immature B cell
B Cell
Hematopoietic stem cell
CFU-Mast
Mast cell
CFU-Bas
Myeloblast
Basophilic Myelocyte
Basophil
CFU-Eo
Eosinophilic Myelocyte
Eosinophil
CFU-M/DC
Myeloid Related Dendritic Cell
Monoblast
Promonocyte
Monocyte
Macrophage
Myeloid Stem Cell
CFU-GEMM
CFU-GM
CFU-G
Neutrophilic Myelocyte
Neutrophil
Bone marrow
BFU-E
CFU-E
Proerythroblast
Erythrocyte
BFU-MK
CFU-MK
Mega-karyocyte
Platelets
Data on KEGG graph
Rendered by Pathview


-1
0
1
JAK-STAT SIGNALING PATHWAY
ECS complex
Ubiquitin mediated proteolysis
Cytokine-cytokine receptor interaction
Cytokine
Hormone
GF
Receptor
JAK
STAM
STAT
STAT
STAT
STAT dimerization
IRF9
TC-PTP
SHP1
SHP2
GRB
SOS
Ras
Raf
MAPK signaling pathway
Proliferation
Differentiation
PI3K
AKT
mTOR
PI3K-AKT signaling pathway
Cell cycle
Cell survival
TC-PTP
PIAS
CBP/P300
SLIM
Proteasome
DNA
CIS
SOCS
Bcl-2
MCL1
Bcl-XL
PIM1
c-Myc
CycD
p21
ACOX
GFAP
Anti-apoptosis
Apoptosis
Cell-cycle progression
Cell-cycle inhibition
Lipid metabolism
Differentiation
Cell cycle
Data on KEGG graph
Rendered by Pathview


-1
0
1
LYSOSOME
bacterium
lysosomal acid hydrolase
Golgi body
transport vesicle
Phagocytosis
phagosome
cytosol
pH~ 7.2
ATP
ADP
acidification regulators
DMXL
WDR7
NCOA7
ATPeV
H+
pH~ 5.0
Transport of synthesized lysosomal enzymes
(See below)
clathrin coat
Endocytosis
early endosome
late endosome
multivesicular body (MVB)
MCOLN1
acid hydrolase
lysosomal membrane protein
lysosome
mitochondria
Autophagy
autophagosome
Regulation of autophagy
Glycosaminoglycan degradation
Other glycan degradation
plasma membrane
Lysosomal acid hydrolases
proteases
cathepsins
napsin
LGMN
TPP1
glycosidases
GLA
GLB
GAA
GBA
IDUA
NAGA
NAGLU
GALC
GUSB
FUCA1
HEXA/B
MANB
LAMAN
NEU1
HYAL1
sulfatases
ARS
GALNS
GNS
IDS
SGSH
lipases
nuclease
phosphatase
LIPA
LYPLA3
DNaseII
ACP2
ACP5
sphingomyelinase
ceramidase
aspartylglucosaminidase
SMPD1
ASAH1
AGA
Other lysosomal enzymes and activators
saposin
GM2A
CLN1
Lysosomal membrane proteins
major lysosomal membrane proteins
LAMP
LIMP
minor lysosomal membrane proteins
NPC
cystinosin
sialin
NRAMP
LAPTM
ABCA2
ABCB9
ACP2
endolyn
LAL/P70
sortilin
CLN3
CLN5
CLN7
HGSNAT
MCOLN1
LITAF
Activation of lysosomal sulfatase precursor
FGE
lysosomal hydrase precursor
from ER
mannose
GNPT
NAGPA
M6P
Snare interactions in vesicular transport
cis Golgi network
Golgi body
trans Golgi network
M6P receptor
MPR
clathrin
AP-1
GGAs
AP-3
AP-4
Receptor-dependent transport
transport vesicle
Receptor recycling
AP-1
M6P
ATPeV
AP-3
lysosome
mannose
mature lysosomal hydrase
late endosome
Transport of synthesized lysosomal enzymes
Data on KEGG graph
Rendered by Pathview


STEROID HORMONE BIOSYNTHESIS
-1
0
1
Steroid biosynthesis
Cholesterol
Cholesterol sulfate
20α-Hydroxy-cholesterol
22β-Hydroxy-cholesterol
20α,22β-Dihydroxy-cholesterol
Pregnenolone
Pregnenolone-sulfate
4-Methylpentanal
17α,20α-Dihydroxy-cholesterol
17α-Hydroxy-pregnenolone
7α-Hydroxy-pregnenolone
20α-Hydroxy-progesterone
Progesterone
11α-Hydroxy-progesterone
17α-Hydroxy-progesterone
17α,20α-Dihydroxy-pregn-4-en-3-one
11-Deoxycorticosterone
11-Deoxycortisol
17α,21-Dihydroxy-pregnenolone
11β,17α,21-Trihydroxy-pregnenolone
Dehydroepiandro-steron sulfate
Dehydro-epiandro-sterone
7α-Hydroxydehydro-epiandrosterone
16α-Hydroxyandrost-4-ene-3,17-dione
3β,17β-Dihydroxy-androst-5-ene
16α-Hydroxydehydro-epiandrosterone
Androst-4-ene-3,17-dione
7α-Hydroxy-androstenedione
7α-Hydroxy-testosterone
Testosterone
19-Hydroxyandrost-4-ene-3,17-dione
19-Oxoandrost-4-ene-3,17-dione
19-Hydroxy-testosterone
19-Oxotestosterone
5β-Androstane-3,17-dione
5α-Androstane-3,17-dione
5α-Dihydro-testosterone
Androsterone
Androsterone-glucuronide
Androstan-3alpha,17beta-diol
5β-Dihydro-testosterone
Testosterone glucuronide
Corticosterone
11-Dehydro-corticosterone
11β,21-Dihydroxy-5β-pregnane-3,20-dione
Tetrahydro-corticosterone
3α,21-Dihydroxy-5β-pregnane-11,20-dione
3α,20α,21-Trihydroxy-5β-pregnane-11-one
Aldosterone hemiacetal
18-Hydroxy-corticosterone
Aldosterone
11β,21-Dihydroxy-3,20-oxo-5β-pregnan-18-al
3α,11β,21-Trihydroxy-20-oxo-5β-pregnan-18-al
21-Hydroxy-5β-pregnane-3,11,20-trione
5α-Dihydro-deoxycorticosterone
Allotetrahydro-deoxycorticosterone
Cortisol
Cortisone
11β-Hydroxyandrost-4-ene-3,17-dione
Adrenosterone
11β,17α,21-Trihydroxy-5β-pregnane-3,20-dione
Urocortisol
Cortolone
Urocortisone
4-Androsten-11beta-ol-3,17-dione
11β-Hydroxytestosterone
5β-Pregnane-3,20-dione
3α-Hydroxy-5β-pregnane-20-one
Pregnanediol
5α-Pregnane-3,20-dione
3α-Hydroxy-5α-pregnan-20-one
5α-Pregnane-3α,20α-diol
5α-Pregnan-20α-ol-3-one
21-Deoxycortisol
Estrone 3-sulfate
Estrone glucuronide
Estradiol-17α
2-Methoxyestrone-3-glucuronide
2-Methoxyestrone
2-Hydroxyestrone
2-Methoxyestrone-3-sulfate
Estrone
16-α-Hydroxyestrone
Estriol
16-Glucuronide-estriol
2-Methoxy-estradiol-17β-3-glucuronide
2-Methoxy-estradiol-17β
2-Hydroxy-estradiol-17β
2-Methoxy-estradiol-17β-3-sulfate
6β-Hydroxy-estradiol-17β
Estradiol-17β
Estradiol-17β-3-glucuronide
Estradiol-17β-3-sulfate
C21-Steroids
C18-Steroids
C19-Steroids
Data on KEGG graph
Rendered by Pathview



>Q. Can you do the same procedure as above to plot the pathview figures for the top 5 down-reguled pathways?

Yes, wee would use the following code and then plot the results:

```
## Focus on top 5 upregulated 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" "hsa04114" "hsa03440"
```

Section 3. Gene Ontology (GO)

```
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
GO:0007156 homophilic cell adhesion       4.892477e-05  3.971899 4.892477e-05
GO:0060429 epithelium development         6.727546e-05  3.834595 6.727546e-05
GO:0007610 behavior                       1.988039e-04  3.557821 1.988039e-04
GO:0048729 tissue morphogenesis           2.470962e-04  3.498983 2.470962e-04
GO:0002009 morphogenesis of an epithelium 3.227439e-04  3.429317 3.227439e-04
GO:0016337 cell-cell adhesion             8.195506e-04  3.163057 8.195506e-04
                                               q.val set.size         exp1
GO:0007156 homophilic cell adhesion       0.1337436      107 4.892477e-05
GO:0060429 epithelium development         0.1337436      478 6.727546e-05
GO:0007610 behavior                       0.2456136      403 1.988039e-04
GO:0048729 tissue morphogenesis           0.2456136      403 2.470962e-04
GO:0002009 morphogenesis of an epithelium 0.2566460      326 3.227439e-04
GO:0016337 cell-cell adhesion             0.3782658      318 8.195506e-04

$less
                                             p.geomean stat.mean        p.val
GO:0000279 M phase                        1.475361e-16 -8.323749 1.475361e-16
GO:0048285 organelle fission              7.498413e-16 -8.160305 7.498413e-16
GO:0000280 nuclear division               2.135098e-15 -8.034814 2.135098e-15
GO:0007067 mitosis                        2.135098e-15 -8.034814 2.135098e-15
GO:0000087 M phase of mitotic cell cycle  5.927567e-15 -7.891758 5.927567e-15
GO:0007059 chromosome segregation         1.055918e-11 -6.988373 1.055918e-11
                                                 q.val set.size         exp1
GO:0000279 M phase                        5.866036e-13      492 1.475361e-16
GO:0048285 organelle fission              1.490684e-12      373 7.498413e-16
GO:0000280 nuclear division               2.122288e-12      349 2.135098e-15
GO:0007067 mitosis                        2.122288e-12      349 2.135098e-15
GO:0000087 M phase of mitotic cell cycle  4.713601e-12      359 5.927567e-15
```

```
GO:0007059 chromosome segregation        6.997217e-09      141 1.055918e-11

$stats
                                          stat.mean     exp1
GO:0007156 homophilic cell adhesion        3.971899 3.971899
GO:0060429 epithelium development          3.834595 3.834595
GO:0007610 behavior                        3.557821 3.557821
GO:0048729 tissue morphogenesis            3.498983 3.498983
GO:0002009 morphogenesis of an epithelium  3.429317 3.429317
GO:0016337 cell-cell adhesion              3.163057 3.163057
```

Section 4. Reactome Analysis

Let's now conduct over-representation enrichment analysis and pathway-topology analysis with Reactome using the previous list of significant genes generated from our differential expression results above.

```
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: 8149"
```

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

> 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 endosomal/vacuolar pathway has the most significant "Entities p-value". The most significant pathways listed do not completely match the previous KEGG reuslts.There may be differences between the two methods because KEGG may put genes under a single pathway while reactome may split them up.