class16_DEGs

Jocelyn Olvera

11/19/2021

```
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
```

1. Data Import

```
colData = read.csv("GSE37704_metadata.csv", row.names=1)
head(colData)
```

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

```
countData = read.csv("GSE37704_featurecounts.csv", 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
```

```
# 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
```

Filter out zeros in the data

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

```
nrow(countData)
```

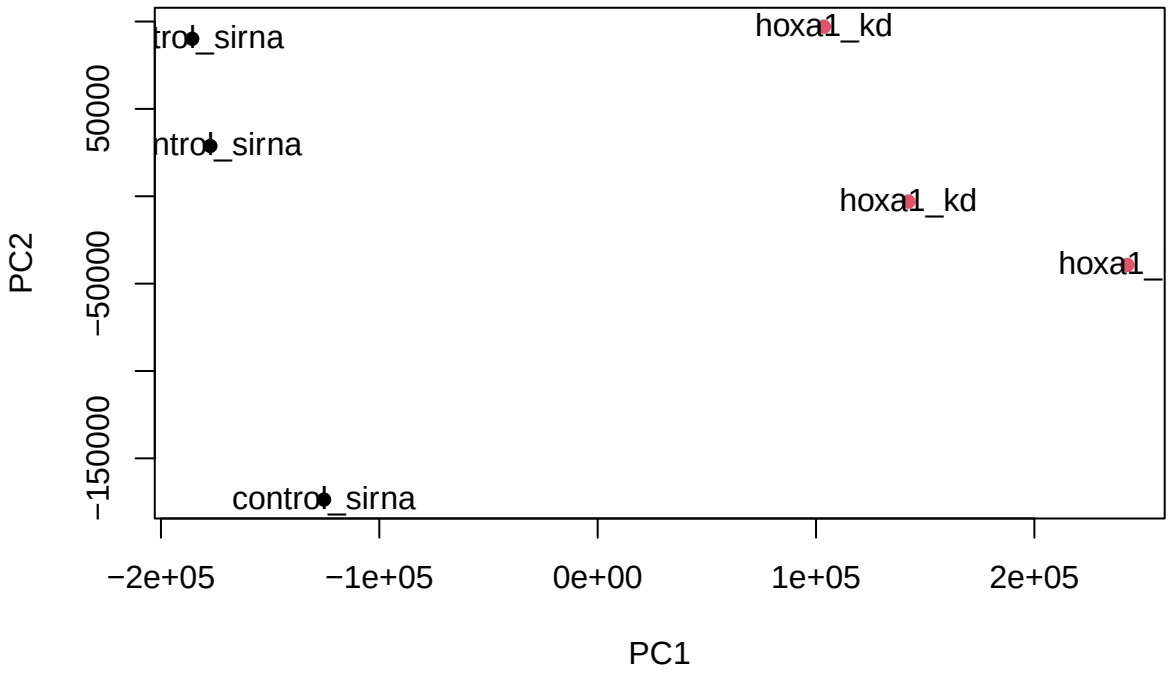
```
## [1] 15975
```

```
pca <- prcomp(t(countData))
summary(pca)
```

```
## Importance of components:
##                             PC1       PC2       PC3       PC4      PC5
## Standard deviation     1.852e+05 1.001e+05 1.998e+04 6.886e+03 5.15e+03
## Proportion of Variance 7.659e-01 2.235e-01 8.920e-03 1.060e-03 5.90e-04
## Cumulative Proportion  7.659e-01 9.894e-01 9.983e-01 9.994e-01 1.00e+00
##                             PC6
## Standard deviation     9.558e-10
## Proportion of Variance 0.000e+00
## Cumulative Proportion  1.000e+00
```

Barry's code

```
plot(pca$x[,1:2], pch =16, col=as.factor(colData$condition))
text(pca$x[,1:2],labels = colData$condition)
```



DESeq Analysis

```
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: 15975 6
## metadata(1): version
## assays(4): counts mu H cooks
## rownames(15975): ENSG00000279457 ENSG00000187634 ... ENSG00000276345
##   ENSG00000271254
## rowData names(22): baseMean baseVar ... deviance maxCooks
## colnames(6): SRR493366 SRR493367 ... SRR493370 SRR493371
## colData names(2): condition sizeFactor
```

```
resultsNames(dds)
```

```
## [1] "Intercept"                           "condition_hoxa1_kd_vs_control_sirna"
```

```
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 15975 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
## ...                   ...            ...       ...        ...         ...
## ENSG00000273748  35.30265       0.674387  0.303666   2.220817 2.63633e-02
## ENSG00000278817   2.42302      -0.388988  1.130394  -0.344117 7.30758e-01
## ENSG00000278384   1.10180       0.332991  1.660261   0.200565 8.41039e-01
## ENSG00000276345  73.64496      -0.356181  0.207716  -1.714752 8.63908e-02
## ENSG00000271254 181.59590      -0.609667  0.141320  -4.314071 1.60276e-05
##                         padj
##                    <numeric>
## ENSG00000279457 6.86555e-01
## ENSG00000187634 5.15718e-03
## ENSG00000188976 1.76549e-35
## ENSG00000187961 1.13413e-07
## ENSG00000187583 9.19031e-01
## ...                     ...
## ENSG00000273748 4.79091e-02
## ENSG00000278817 8.09772e-01
## ENSG00000278384 8.92654e-01
## ENSG00000276345 1.39762e-01
## ENSG00000271254 4.53648e-05
```

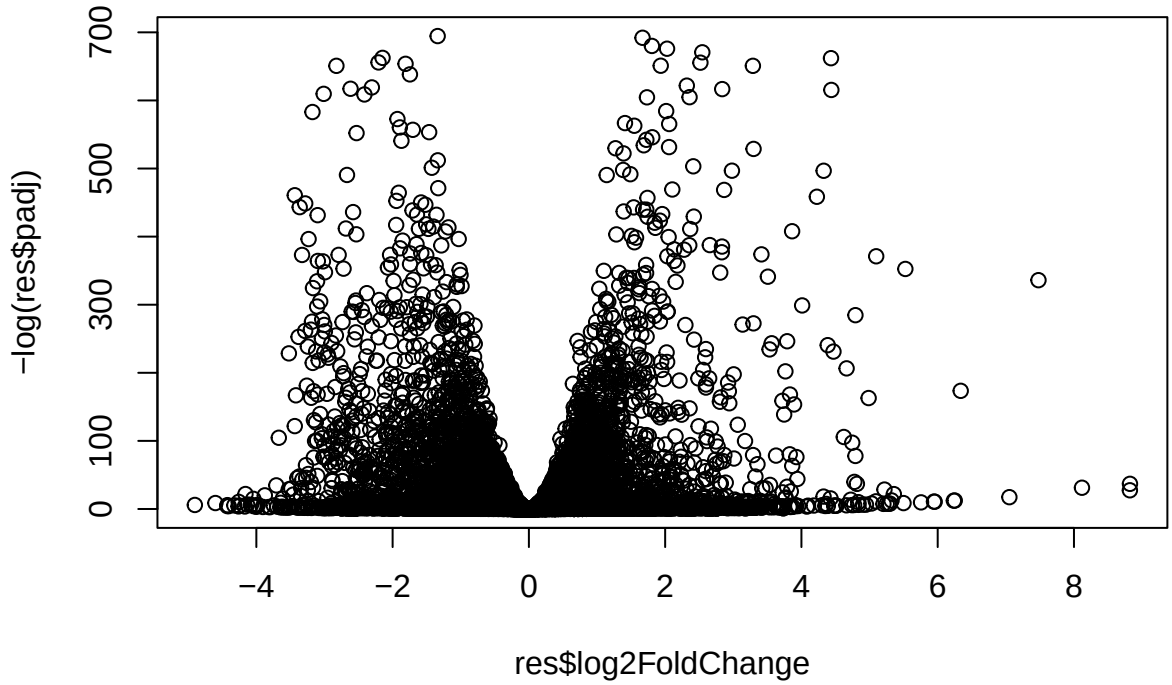
Q. Call the summary() function on your results to get a sense of how many genes are up or down-regulated at the default 0.1 p-value cutoff.

```
summary(res)
```

```
##
## out of 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
```

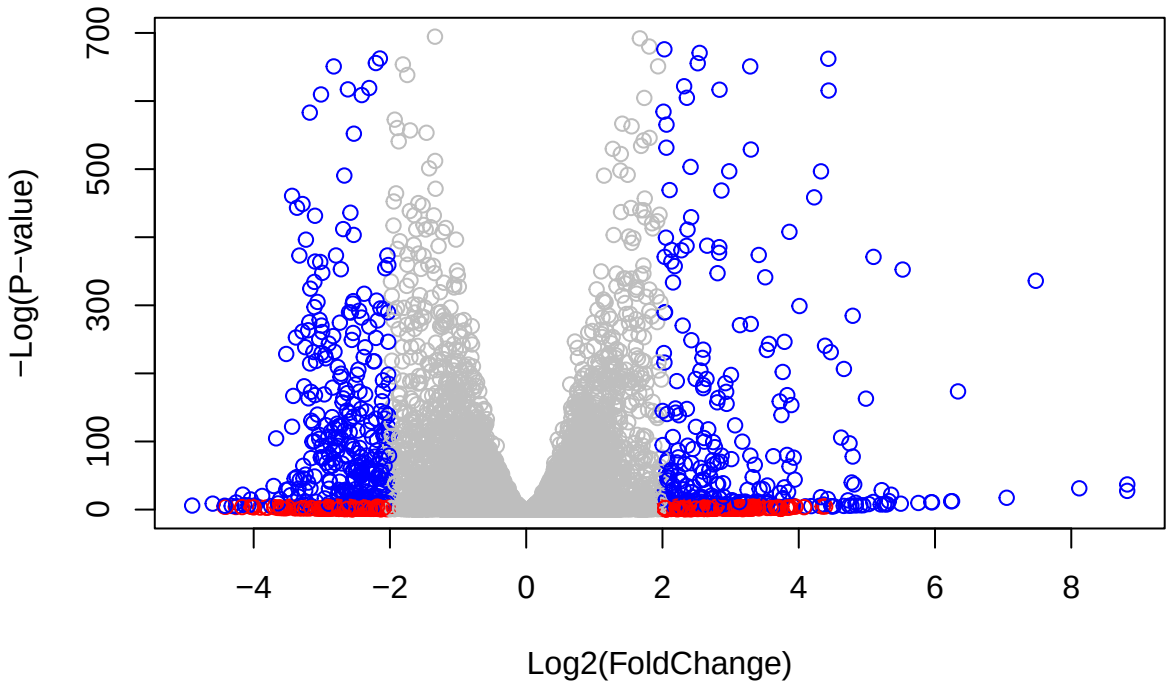
#4. Volcano Plot

```
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(P-value)" )
```



Gene Annotation

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

```
head(rownames(res))
```

```
## [1] "ENSG00000279457" "ENSG00000187634" "ENSG00000188976" "ENSG00000187961"
## [5] "ENSG00000187583" "ENSG00000187642"
```

```
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="ENTREZID",
                    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="ENTREZID",
                    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   102723897   102723897   102723897
## ENSG00000187634 5.15718e-03      148398      148398      148398
## ENSG00000188976 1.76549e-35       26155       26155       26155
## ENSG00000187961 1.13413e-07      339451      339451      339451
## ENSG00000187583 9.19031e-01       84069       84069       84069
## ENSG00000187642 4.03379e-01       84808       84808       84808
## ENSG00000188290 1.30538e-24       57801       57801       57801
## ENSG00000187608 2.37452e-02        9636        9636        9636
## ENSG00000188157 4.21963e-16      375790      375790      375790
## ENSG00000237330          NA      401934      401934      401934
```

```
ord <- order( res$padj )
#View(res[ord,])
head(res[ord,])
```

```
## 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 9 columns
##                   baseMean log2FoldChange     lfcSE      stat    pvalue
##                  <numeric>      <numeric> <numeric> <numeric> <numeric>
## ENSG00000117519   4483.63       -2.42272 0.0600016  -40.3776         0
## ENSG00000183508   2053.88        3.20196 0.0724172   44.2154         0
## ENSG00000159176   5692.46       -2.31374 0.0575534  -40.2016         0
```

```
## ENSG00000150938   7442.99       -2.05963 0.0538449  -38.2512         0
## ENSG00000116016   4423.95       -1.88802 0.0431680  -43.7366         0
## ENSG00000136068   3796.13       -1.64979 0.0439354  -37.5504         0
##                      padj      symbol      entrez        name
##                 <numeric> <character> <character> <character>
## ENSG00000117519         0        1266        1266        1266
## ENSG00000183508         0       54855       54855       54855
## ENSG00000159176         0        1465        1465        1465
## ENSG00000150938         0       51232       51232       51232
## ENSG00000116016         0        2034        2034        2034
## ENSG00000136068         0        2317        2317        2317
```

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[ord,],"deseq_results.csv")
```

Section 2. Pathway Analysis

```
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"
```

```
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
```

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

```
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                   8.995727e-06 -4.378644 8.995727e-06
## hsa03030 DNA replication              9.424076e-05 -3.951803 9.424076e-05
## hsa03013 RNA transport                1.375901e-03 -3.028500 1.375901e-03
## hsa03440 Homologous recombination     3.066756e-03 -2.852899 3.066756e-03
## hsa04114 Oocyte meiosis               3.784520e-03 -2.698128 3.784520e-03
## hsa00010 Glycolysis / Gluconeogenesis 8.961413e-03 -2.405398 8.961413e-03
##                                            q.val set.size         exp1
```

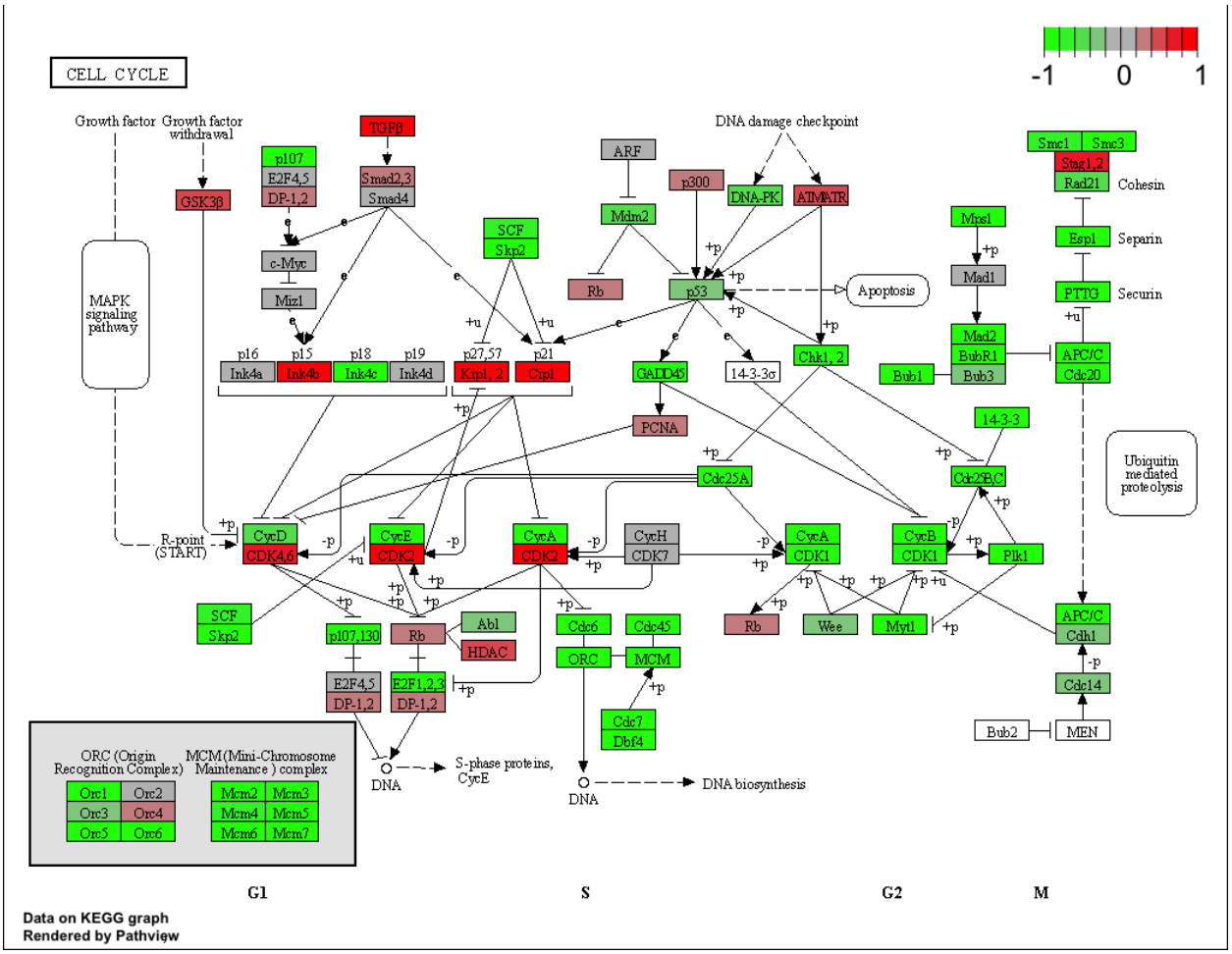
```
## hsa04110 Cell cycle                    0.001448312      121 8.995727e-06
## hsa03030 DNA replication               0.007586381       36 9.424076e-05
## hsa03013 RNA transport                 0.073840037      144 1.375901e-03
## hsa03440 Homologous recombination      0.121861535       28 3.066756e-03
## hsa04114 Oocyte meiosis                0.121861535      102 3.784520e-03
## hsa00010 Glycolysis / Gluconeogenesis 0.212222694       53 8.961413e-03
```

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

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

```
## Info: Working in directory /Users/jocelynolvera/Documents/phd_yr1_2021/bggn213/bggn213_Rstudio_git/bl
```

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




```
# A different PDF based output of the same data
pathview(gene.data=foldchanges, pathway.id="hsa04110", kegg.native=FALSE)
```

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

```
## Info: Working in directory /Users/jocelynolvera/Documents/phd_yr1_2021/bggn213/bggn213_Rstudio_git/bl
```

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

```
## 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" "hsa00140" "hsa04142" "hsa04330"
```

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

```
## 'select()' returned 1:1 mapping between keys and columns
## Info: Working in directory /Users/jocelynolvera/Documents/phd_yr1_2021/bggn213/bggn213_Rstudio_git/bl
## Info: Writing image file hsa04640.pathview.png
## 'select()' returned 1:1 mapping between keys and columns
## Info: Working in directory /Users/jocelynolvera/Documents/phd_yr1_2021/bggn213/bggn213_Rstudio_git/bl
## Info: Writing image file hsa04630.pathview.png
## 'select()' returned 1:1 mapping between keys and columns
## Info: Working in directory /Users/jocelynolvera/Documents/phd_yr1_2021/bggn213/bggn213_Rstudio_git/bl
## Info: Writing image file hsa00140.pathview.png
## 'select()' returned 1:1 mapping between keys and columns
## Info: Working in directory /Users/jocelynolvera/Documents/phd_yr1_2021/bggn213/bggn213_Rstudio_git/bl
## 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/jocelynolvera/Documents/phd_yr1_2021/bggn213/bggn213_Rstudio_git/bl
## Info: Writing image file hsa04330.pathview.png
```

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

```
# A different PDF based output of the same data
pathview(gene.data=foldchanges, pathway.id="hsa04110", kegg.native=FALSE)
```

```
## 'select()' returned 1:1 mapping between keys and columns
## Info: Working in directory /Users/jocelynolvera/Documents/phd_yr1_2021/bggn213/bggn213_Rstudio_git/bl
## Info: Writing image file hsa04110.pathview.pdf
```

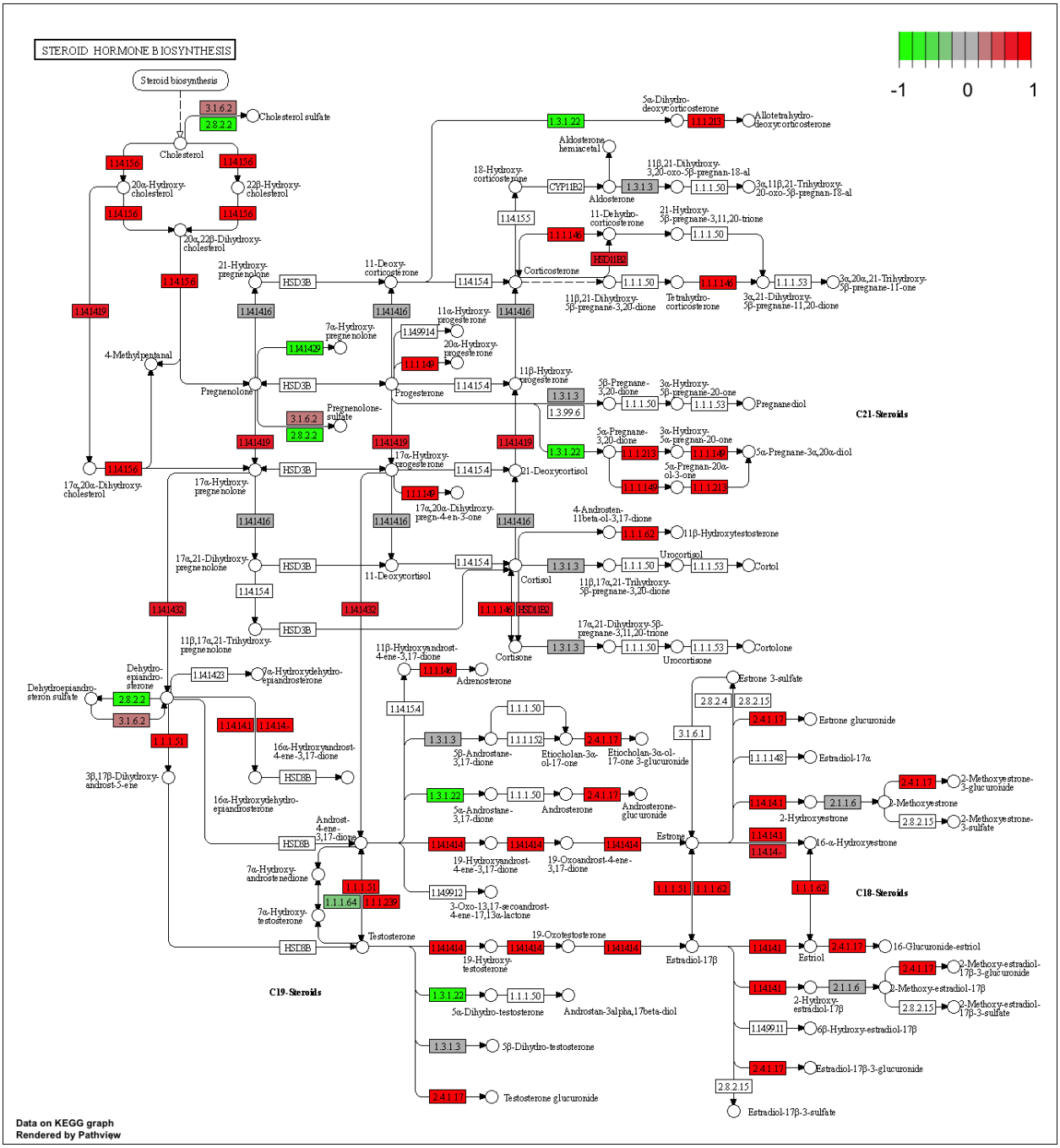
```
## Focus on top 5 upregulated pathways here for demo purposes only
keggrespathways <- rownames(keggres$greater)[1:5]

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

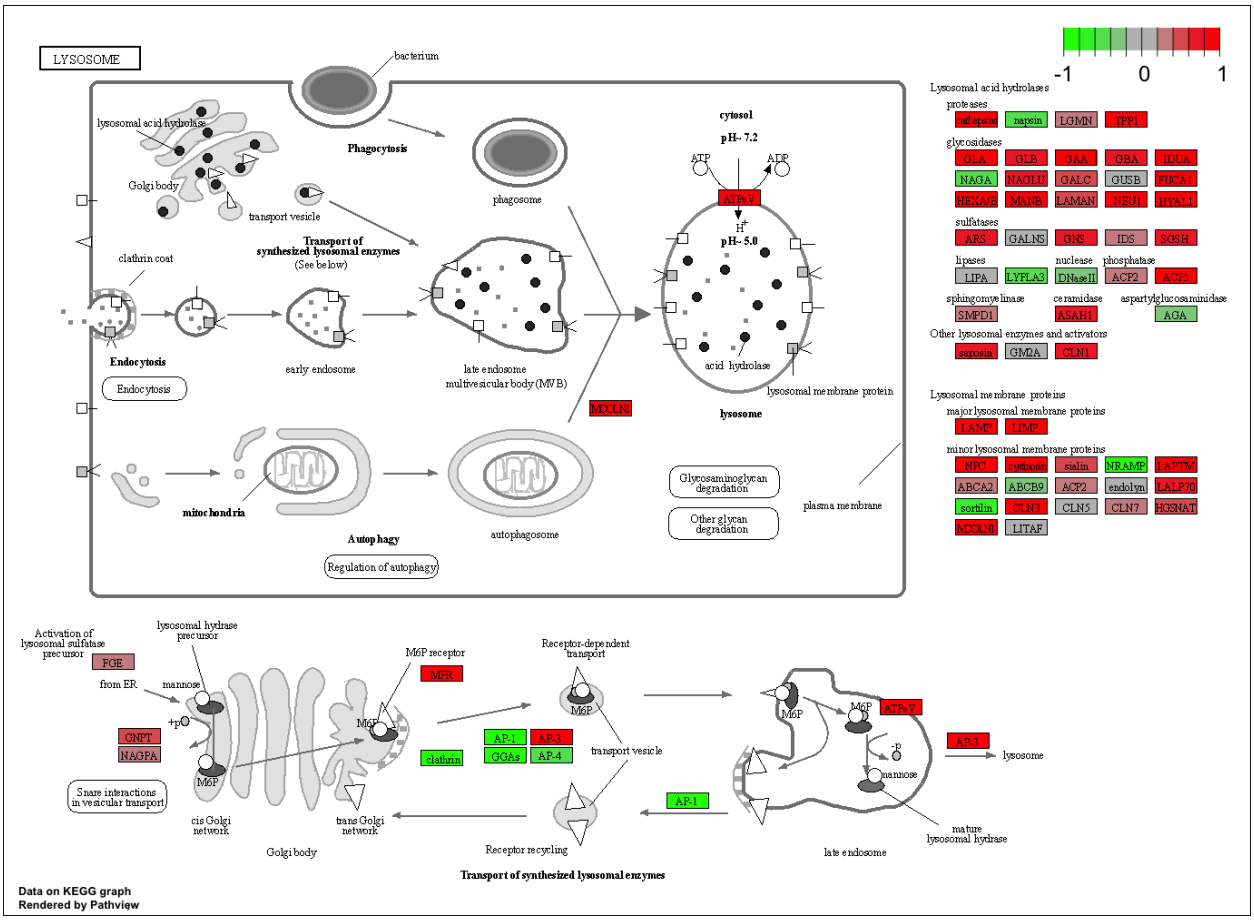
```
## [1] "hsa04640" "hsa04630" "hsa00140" "hsa04142" "hsa04330"
```

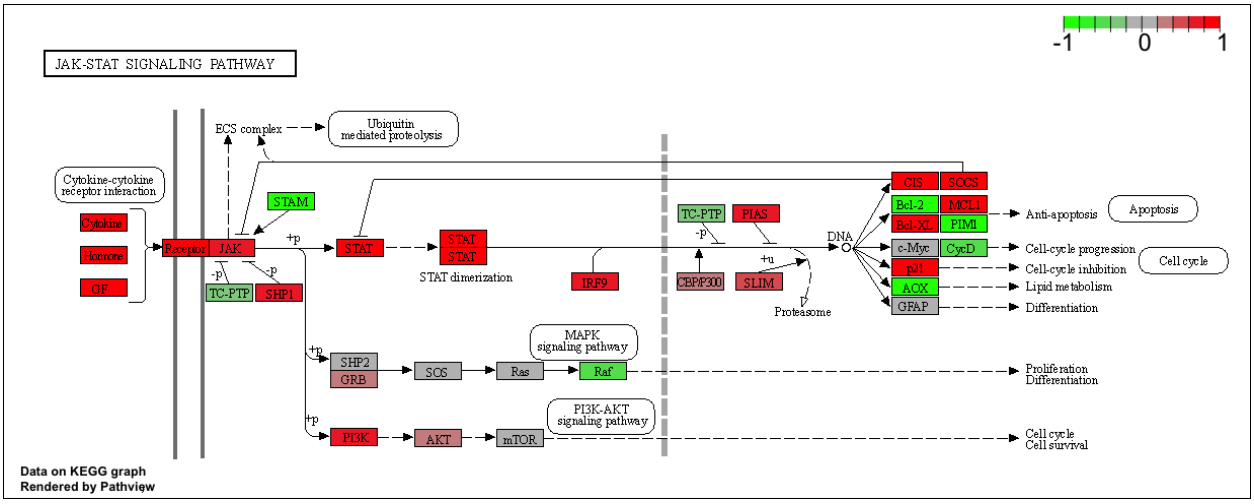
```
pathview(gene.data=foldchanges, pathway.id=keggresids, species="hsa")
```

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

```
## Info: Working in directory /Users/jocelynolvera/Documents/phd_yr1_2021/bggn213/bggn213_Rstudio_git/bl
## Info: Writing image file hsa04640.pathview.png
## 'select()' returned 1:1 mapping between keys and columns
## Info: Working in directory /Users/jocelynolvera/Documents/phd_yr1_2021/bggn213/bggn213_Rstudio_git/bl
## Info: Writing image file hsa04630.pathview.png
## 'select()' returned 1:1 mapping between keys and columns
## Info: Working in directory /Users/jocelynolvera/Documents/phd_yr1_2021/bggn213/bggn213_Rstudio_git/bl
## Info: Writing image file hsa00140.pathview.png
## 'select()' returned 1:1 mapping between keys and columns
## Info: Working in directory /Users/jocelynolvera/Documents/phd_yr1_2021/bggn213/bggn213_Rstudio_git/bl
## 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/jocelynolvera/Documents/phd_yr1_2021/bggn213/bggn213_Rstudio_git/bl
## Info: Writing image file hsa04330.pathview.png
```


STEROID HORMONE BIOSYNTHESIS
-1
0
1
Steroid biosynthesis
Cholesterol
Cholesterol sulfate
20α-Hydroxy-cholesterol
22β-Hydroxy-cholesterol
20α,22β-Dihydroxy-cholesterol
21-Hydroxy-pregnenolone
11-Deoxy-corticosterone
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
11α-Hydroxy-progesterone
7α-Hydroxy-pregnenolone
20α-Hydroxy-progesterone
4-Methylpentanal
Pregnenolone
Progesterone
11β-Hydroxy-progesterone
Pregnenolone-sulfate
5β-Pregnane-3,20-dione
3α-Hydroxy-5β-pregnane-20-one
Pregnanediol
C21-Steroids
17α,20α-Dihydroxy-cholesterol
17α-Hydroxy-pregnenolone
17α-Hydroxy-progesterone
21-Deoxycortisol
17α,20α-Dihydroxy-pregn-4-en-3-one
5α-Pregnane-3,20-dione
3α-Hydroxy-5α-pregnan-20-one
5α-Pregnane-3α,20α-diol
5α-Pregnan-20α-ol-3-one
4-Androsten-11beta-ol-3,17-dione
11β-Hydroxytestosterone
17α,21-Dihydroxy-pregnenolone
11-Deoxycortisol
Cortisol
Urocortisol
Cortol
11β,17α,21-Trihydroxy-5β-pregnane-3,20-dione
11β,17α,21-Trihydroxy-pregnenolone
17α,21-Dihydroxy-5β-pregnane-3,11,20-trione
Cortisone
Urocortisone
Cortolone
11β-Hydroxyandrost-4-ene-3,17-dione
Adrenosterone
Dehydroepiandro-sterone sulfate
Dehydro-epiandro-sterone
7α-Hydroxydehydro-epiandrosterone
16α-Hydroxydehydro-epiandrosterone
3β,17β-Dihydroxy-androst-5-ene
16α-Hydroxyandrost-4-ene-3,17-dione
Androst-4-ene-3,17-dione
5β-Androstane-3,17-dione
Etiocholan-3α-ol-17-one
Etiocholan-3α-ol-17-one 3-glucuronide
5α-Androstane-3,17-dione
Androsterone
Androsterone-glucuronide
19-Hydroxyandrost-4-ene-3,17-dione
19-Oxoandrost-4-ene-3,17-dione
Estrone
Estrone 3-sulfate
Estrone glucuronide
Estradiol-17α
2-Hydroxyestrone
2-Methoxyestrone
2-Methoxyestrone-3-glucuronide
2-Methoxyestrone-3-sulfate
16α-Hydroxyestrone
7α-Hydroxy-androstenedione
7α-Hydroxy-testosterone
3-Oxo-13,17-secoandrost-4-ene-17,13α-lactone
Testosterone
19-Oxotestosterone
19-Hydroxy-testosterone
Estradiol-17β
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-3-glucuronide
Estradiol-17β-3-sulfate
C18-Steroids
C19-Steroids
5α-Dihydro-testosterone
Androstan-3alpha,17beta-diol
5β-Dihydro-testosterone
Testosterone glucuronide
Data on KEGG graph
Rendered by Pathview


LYSOSOME
bacterium
lysosomal acid hydrolase
Golgi body
transport vesicle
Phagocytosis
phagosome
cytosol
pH~ 7.2
ATP
ADP
ATPeV
H+
pH~ 5.0
acid hydrolase
lysosomal membrane protein
lysosome
Transport of synthesized lysosomal enzymes
(See below)
clathrin coat
Endocytosis
Endocytosis
early endosome
late endosome
multivesicular body (MVB)
MCOLN1
mitochondria
Autophagy
autophagosome
Regulation of autophagy
Glycosaminoglycan degradation
Other glycan degradation
plasma membrane
-1
0
1
Lysosomal acid hydrolases
proteases
cathepsin
pepsin
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
LALP70
sortilin
CLN3
CLN5
CLN7
HGSNAT
MCOLN1
LITAF
Activation of lysosomal sulfatase precursor
FGE
lysosomal hydrase precursor
from ER
mannose
+p
GNPT
NAGPA
M6P
Snare interactions in vesicular transport
cis Golgi network
trans Golgi network
Golgi body
M6P receptor
MPR
clathrin
AP-1
AP-3
GGAs
AP-4
Receptor-dependent transport
transport vesicle
Receptor recycling
AP-1
M6P
ATPeV
-p
mannose
AP-3
lysosome
mature lysosomal hydrase
late endosome
Transport of synthesized lysosomal enzymes
Data on KEGG graph
Rendered by Pathview


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