

Class16 RNAseq Mini Project

Saumya Ranjan

11/18/2021

##Section 1. Differential Expression Analysis

```
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 <- "GSE37704_metadata.csv"
```

```
countFile <- "GSE37704_featurecounts.csv"
```

```
# 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
```

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

```
# Note we need to remove the odd first $length col
countData2 <- as.matrix(countData[,-1])
head(countData2)
```

```
##                      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
```

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.
countData3 = countData2[rowSums(countData2)>1, ]
head(countData3)
```

```
##                      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
```

#Running DESeq2

```
library(DESeq2)
dds = DESeqDataSetFromMatrix(countData = countData3,
                             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
```

```
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 15280 rows and 6 columns
```

```
##           baseMean log2FoldChange    lfcSE      stat      pvalue
##           <numeric>    <numeric> <numeric> <numeric> <numeric>
## ENSG00000279457   29.9136    0.1792570 0.3248225   0.551861 5.81043e-01
## ENSG00000187634  183.2296    0.4264571 0.1402660   3.040345 2.36307e-03
## ENSG00000188976 1651.1881   -0.6927205 0.0548462 -12.630233 1.43852e-36
## ENSG00000187961  209.6379    0.7297556 0.1318601   5.534318 3.12441e-08
## ENSG00000187583   47.2551    0.0405766 0.2718936   0.149237 8.81367e-01
## ...           ...           ...      ...      ...      ...
## ENSG00000273748   35.30265    0.674387 0.303667   2.220811 2.63638e-02
## ENSG00000278817    2.42302   -0.388988 1.130397  -0.344117 7.30759e-01
## ENSG00000278384    1.10180    0.332991 1.660266   0.200565 8.41039e-01
## ENSG00000276345   73.64496   -0.356181 0.207716  -1.714748 8.63915e-02
## ENSG00000271254  181.59590   -0.609667 0.141321  -4.314064 1.60281e-05
##           padj
##           <numeric>
## ENSG00000279457 6.85034e-01
## ENSG00000187634 5.14047e-03
## ENSG00000188976 1.75806e-35
## ENSG00000187961 1.13048e-07
## ENSG00000187583 9.19159e-01
## ...           ...
## ENSG00000273748 4.77539e-02
## ENSG00000278817 8.08838e-01
## ENSG00000278384 8.92833e-01
## ENSG00000276345 1.39307e-01
## ENSG00000271254 4.52184e-05
```

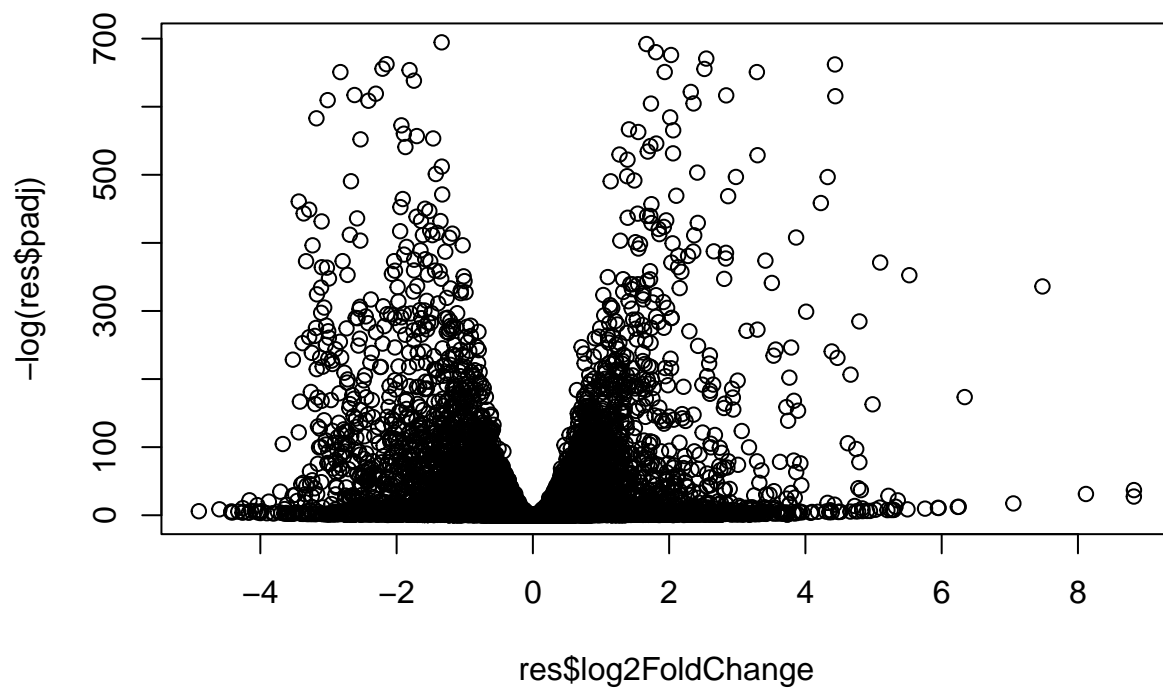
Q. Call the `summary()` function on your results to get a sense of how many genes are up or down-regulated at the default 0.1 p-value cutoff.

```
res0.1 <- results(dds, alpha=0.1)
summary(res0.1)
```

```
##
## 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
```

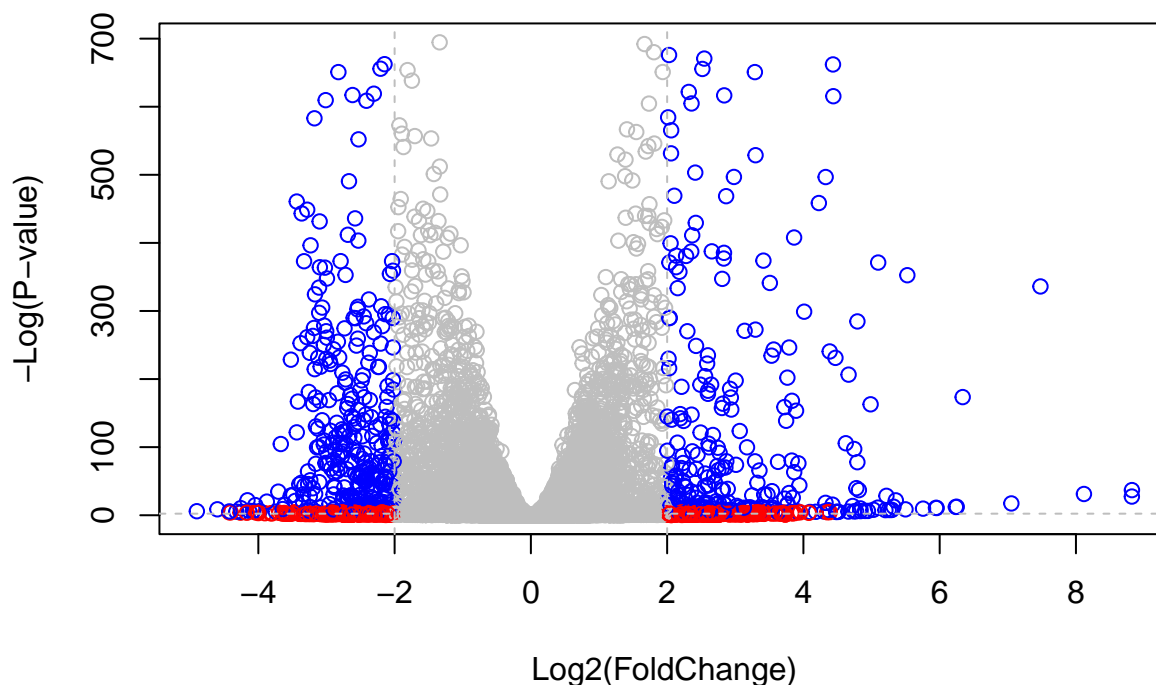
```
#volcano plot
```

```
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))
mycols[ abs(res$log2FoldChange) > 2 ] <- "red"
inds <- (res$padj < 0.01) & (abs(res$log2FoldChange) > 2 )
mycols[ inds ] <- "blue"
# Volcano plot with custom colors
plot( res$log2FoldChange, -log(res$padj),
      col=mycols, ylab="-Log(P-value)", xlab="Log2(FoldChange)" )
# Cut-off lines
abline(v=c(-2,2), col="gray", lty=2)
abline(h=-log(0.1), col="gray", lty=2)
```



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      0.1792570 0.3248225  0.551861 5.81043e-01
## ENSG00000187634 183.2296      0.4264571 0.1402660  3.040345 2.36307e-03
## ENSG00000188976 1651.1881    -0.6927205 0.0548462 -12.630233 1.43852e-36
## ENSG00000187961 209.6379      0.7297556 0.1318601  5.534318 3.12441e-08
## ENSG00000187583  47.2551      0.0405766 0.2718936  0.149237 8.81367e-01
## ENSG00000187642  11.9798      0.5428107 0.5215615  1.040742 2.97995e-01
## ENSG00000188290 108.9221      2.0570639 0.1969057 10.446951 1.51312e-25
## ENSG00000187608 350.7169      0.2573837 0.1027266  2.505522 1.22271e-02
## ENSG00000188157 9128.4394     0.3899088 0.0467158  8.346400 7.03753e-17
## ENSG00000131591 156.4791      0.1965923 0.1456112  1.350118 1.76978e-01
##      padj      symbol      entrez      name
##      <numeric> <character> <character> <character>
## ENSG00000279457 6.85034e-01  WASH9P  102723897 WAS protein family h..
## ENSG00000187634 5.14047e-03  SAMD11  148398  sterile alpha motif ..
## ENSG00000188976 1.75806e-35  NOC2L   26155  NOC2 like nucleolar ..
## ENSG00000187961 1.13048e-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 1.30139e-24  HES4    57801  hes family bHLH tran..
## ENSG00000187608 2.36679e-02  ISG15   9636  ISG15 ubiquitin like..
## ENSG00000188157 4.20249e-16  AGRN    375790 agrin
## ENSG00000131591 2.60894e-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

```

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                1.003993e-05 -4.353454 1.003993e-05
## hsa03030 DNA replication            8.909556e-05 -3.968611 8.909556e-05
## hsa03013 RNA transport              1.470984e-03 -3.007794 1.470984e-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 6.059195e-03 -2.558327 6.059195e-03
##                                     q.val set.size      exp1
## hsa04110 Cell cycle                0.001606389      120 1.003993e-05
## hsa03030 DNA replication            0.007127645       36 8.909556e-05
## hsa03013 RNA transport              0.077876206     143 1.470984e-03
## hsa04114 Oocyte meiosis            0.077876206      99 1.946905e-03
## hsa03440 Homologous recombination  0.094143660      28 2.941989e-03
## hsa00010 Glycolysis / Gluconeogenesis 0.161578526      48 6.059195e-03

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

## 'select()' returned 1:1 mapping between keys and columns
## Info: Working in directory /home/saranjan/Class16
## 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 /home/saranjan/Class16
## Info: Writing image file hsa04110.pathview.pdf
## Focus on top 5 upregulated pathways here for demo purposes only
keggrespathways <- rownames(keggres$greater)[1:5]

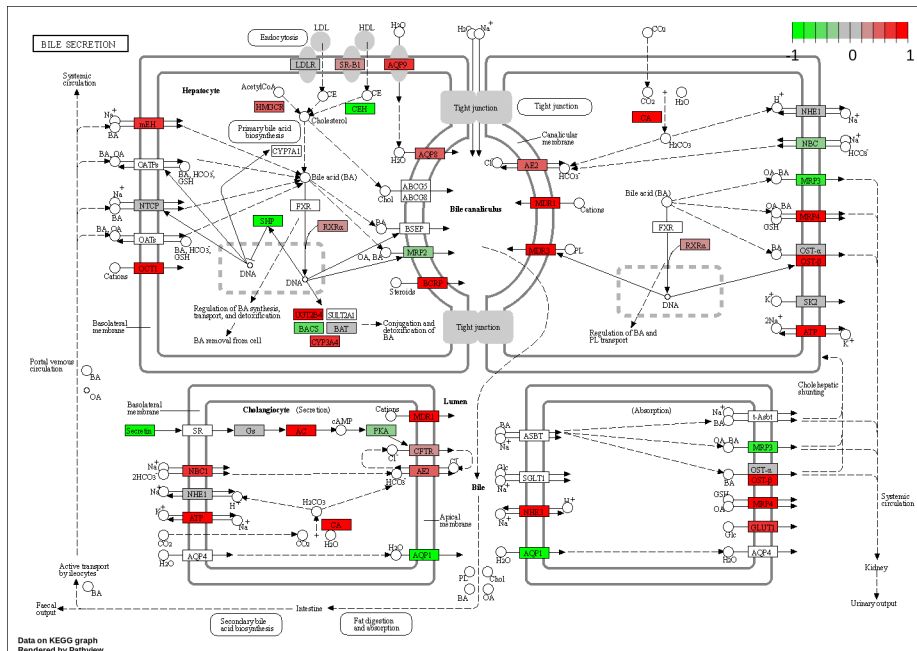
# Extract the 8 character long IDs part of each string
keggresids = substr(keggrespathways, start=1, stop=8)
keggresids

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

```

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

```
## 'select()' returned 1:1 mapping between keys and columns
## Info: Working in directory /home/saranjan/Class16
## Info: Writing image file hsa04640.pathview.png
## 'select()' returned 1:1 mapping between keys and columns
## Info: Working in directory /home/saranjan/Class16
## Info: Writing image file hsa04630.pathview.png
## 'select()' returned 1:1 mapping between keys and columns
## Info: Working in directory /home/saranjan/Class16
## 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 /home/saranjan/Class16
## Info: Writing image file hsa00140.pathview.png
## 'select()' returned 1:1 mapping between keys and columns
## Info: Working in directory /home/saranjan/Class16
## Info: Writing image file hsa04740.pathview.png
## Info: some node width is different from others, and hence adjusted!
```



```
pca <-prcomp(t(countData3))
```

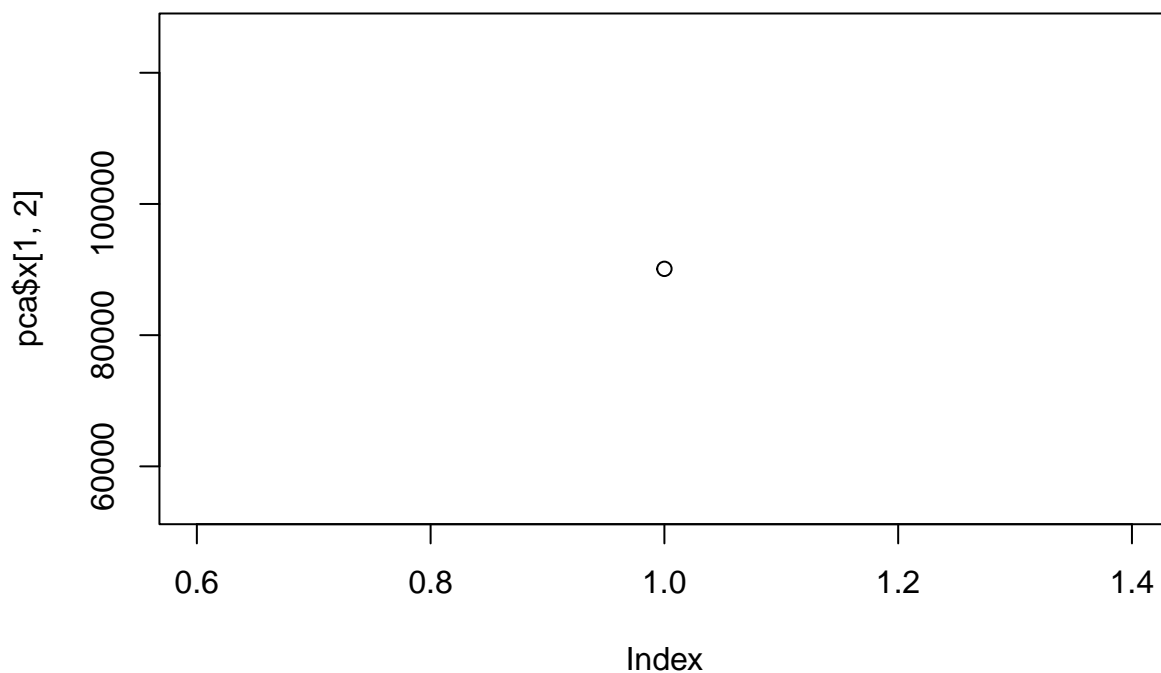
```
plot(pca$x[1,2], cols=mycols)
```

```
## Warning in plot.window(...): "cols" is not a graphical parameter
```

```
## Warning in plot.xy(xy, type, ...): "cols" is not a graphical parameter
## Warning in axis(side = side, at = at, labels = labels, ...): "cols" is not a
## graphical parameter

## Warning in axis(side = side, at = at, labels = labels, ...): "cols" is not a
## graphical parameter

## Warning in box(...): "cols" is not a graphical parameter
## Warning in title(...): "cols" is not a graphical parameter
```



Gene ontology

```
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
## G0:0007156 homophilic cell adhesion 4.892475e-05 3.971899 4.892475e-05
## G0:0060429 epithelium development 6.727544e-05 3.834595 6.727544e-05
## G0:0007610 behavior 1.988040e-04 3.557821 1.988040e-04
```

```

## G0:0048729 tissue morphogenesis      2.470961e-04  3.498983 2.470961e-04
## G0:0002009 morphogenesis of an epithelium 3.227438e-04  3.429317 3.227438e-04
## G0:0016337 cell-cell adhesion      8.195510e-04  3.163057 8.195510e-04
##                                     q.val set.size      exp1
## G0:0007156 homophilic cell adhesion    0.1337436      107 4.892475e-05
## G0:0060429 epithelium development      0.1337436      478 6.727544e-05
## G0:0007610 behavior                    0.2456135      403 1.988040e-04
## G0:0048729 tissue morphogenesis      0.2456135      403 2.470961e-04
## G0:0002009 morphogenesis of an epithelium 0.2566459      326 3.227438e-04
## G0:0016337 cell-cell adhesion      0.3782657      318 8.195510e-04
##
## $less
##                                     p.geomean stat.mean      p.val
## G0:0000279 M phase                    1.475361e-16 -8.323749 1.475361e-16
## G0:0048285 organelle fission           7.498414e-16 -8.160305 7.498414e-16
## G0:0000280 nuclear division            2.135098e-15 -8.034814 2.135098e-15
## G0:0007067 mitosis                    2.135098e-15 -8.034814 2.135098e-15
## G0:0000087 M phase of mitotic cell cycle 5.927566e-15 -7.891758 5.927566e-15
## G0:0007059 chromosome segregation      1.055918e-11 -6.988373 1.055918e-11
##                                     q.val set.size      exp1
## G0:0000279 M phase                    5.866037e-13      492 1.475361e-16
## G0:0048285 organelle fission           1.490685e-12      373 7.498414e-16
## G0:0000280 nuclear division            2.122288e-12      349 2.135098e-15
## G0:0007067 mitosis                    2.122288e-12      349 2.135098e-15
## G0:0000087 M phase of mitotic cell cycle 4.713601e-12      359 5.927566e-15
## G0:0007059 chromosome segregation      6.997219e-09      141 1.055918e-11
##
## $stats
##                                     stat.mean      exp1
## G0:0007156 homophilic cell adhesion    3.971899 3.971899
## G0:0060429 epithelium development      3.834595 3.834595
## G0:0007610 behavior                    3.557821 3.557821
## G0:0048729 tissue morphogenesis      3.498983 3.498983
## G0:0002009 morphogenesis of an epithelium 3.429317 3.429317
## G0:0016337 cell-cell adhesion      3.163057 3.163057

```