

Class16: RNASeq Mini Project

Yuhan Zhang (PID: A13829264)

11/19/2021

```
library(DESeq2)
```

1. Data Import

Load data:

```
metaFile <- "GSE37704_metadata.csv"
countFile <- "GSE37704_featurecounts.csv"
```

```
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
```

```
countData <- read.csv(countFile, row.names = 1)
head(countData)
```

```
##           length SRR493366 SRR493367 SRR493368 SRR493369 SRR493370
## ENSG00000186092    918         0         0         0         0
## ENSG00000279928    718         0         0         0         0
## ENSG00000279457   1982        23        28        29        28
## ENSG00000278566    939         0         0         0         0
## ENSG00000273547    939         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 to remove the first column (i.e. `countData$length`) to match with metadata:

```
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
```

We also need to remove entries that has no reading (0 across all columns)

```
row.rm = rowSums(countData) != 0
countData <- countData[row.rm,]
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
```

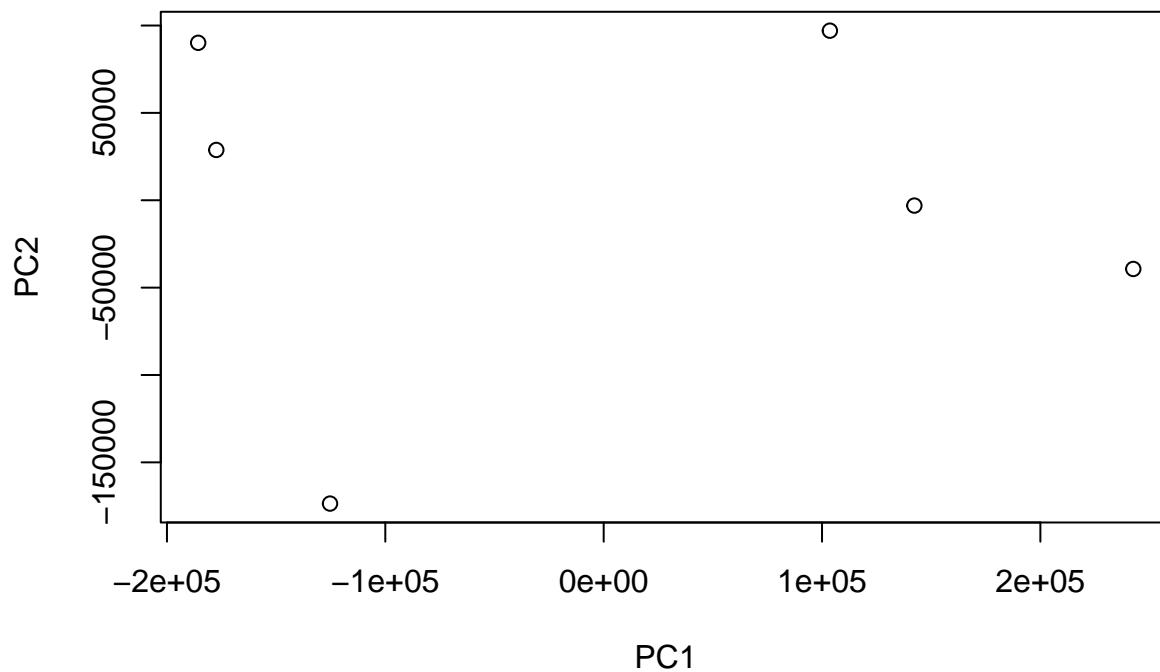
2. PCA for Quality Control

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

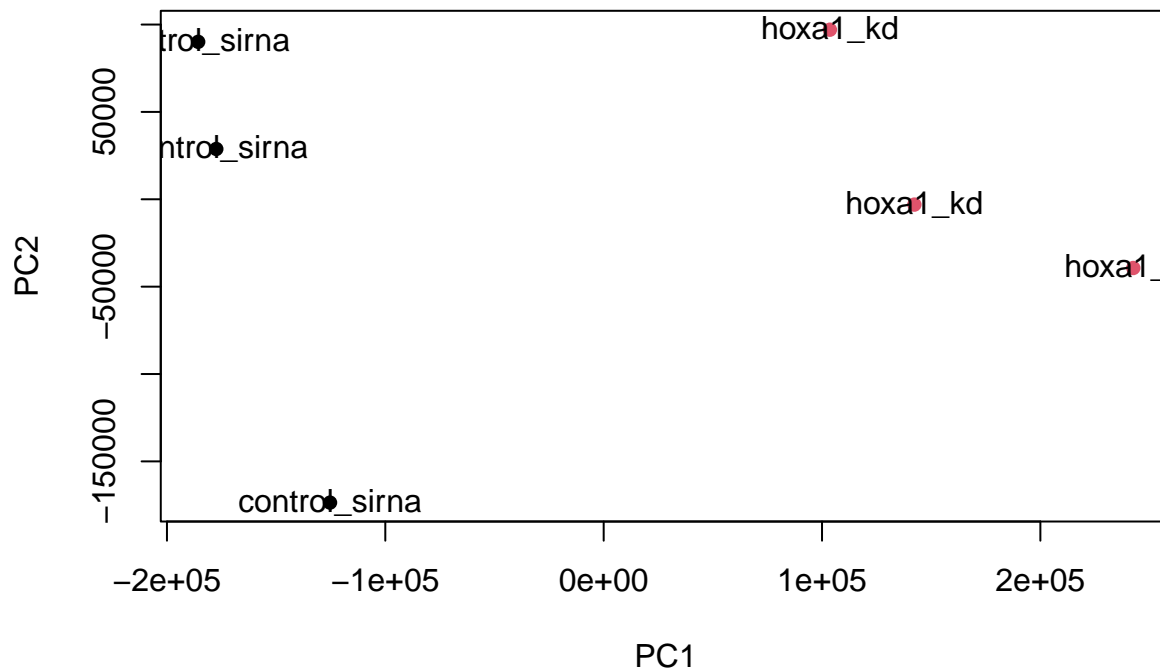
```
## Importance of components:
##          PC1          PC2          PC3          PC4          PC5
## Standard deviation 1.852e+05 1.001e+05 1.998e+04 6.886e+03 5.15e+03
## Proportion of Variance 7.659e-01 2.235e-01 8.920e-03 1.060e-03 5.90e-04
## Cumulative Proportion 7.659e-01 9.894e-01 9.983e-01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

Plot first and second:

```
plot(pca$x)
```



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

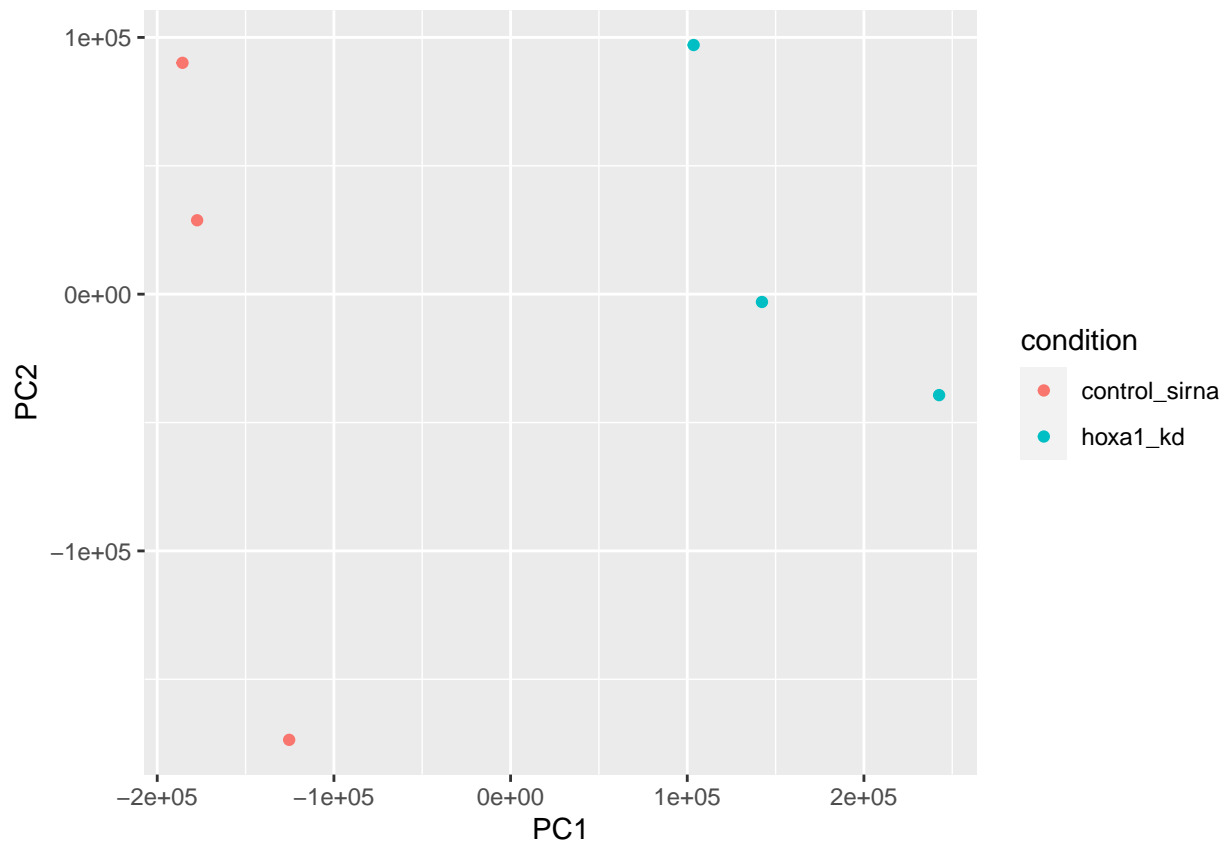


ggplot version:

```
library(ggplot2)

x <- as.data.frame(pca$x)
x$condition <- colData$condition

ggplot(x, aes(PC1, PC2, col=condition)) +
  geom_point()
```



3. Running DESeq2

```
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
```

Get result from our DESeq data:

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

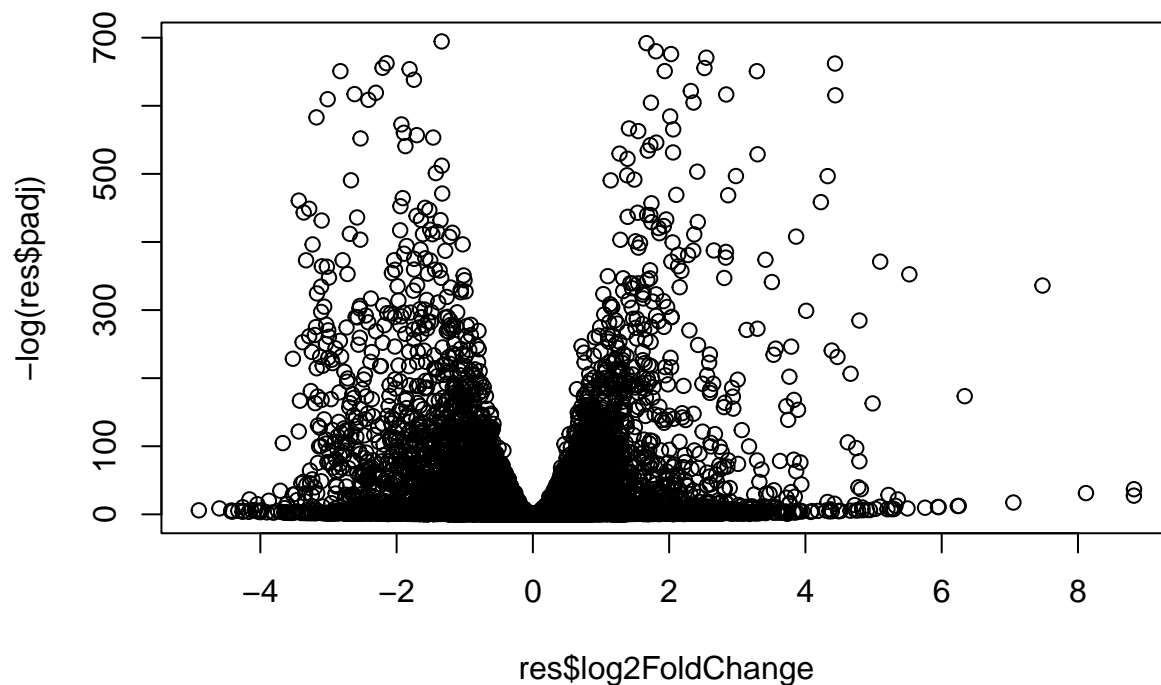
```
summary(res)
```

```
##
## 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

Let's do the classic log2-FoldChange vs p-value volcano plot

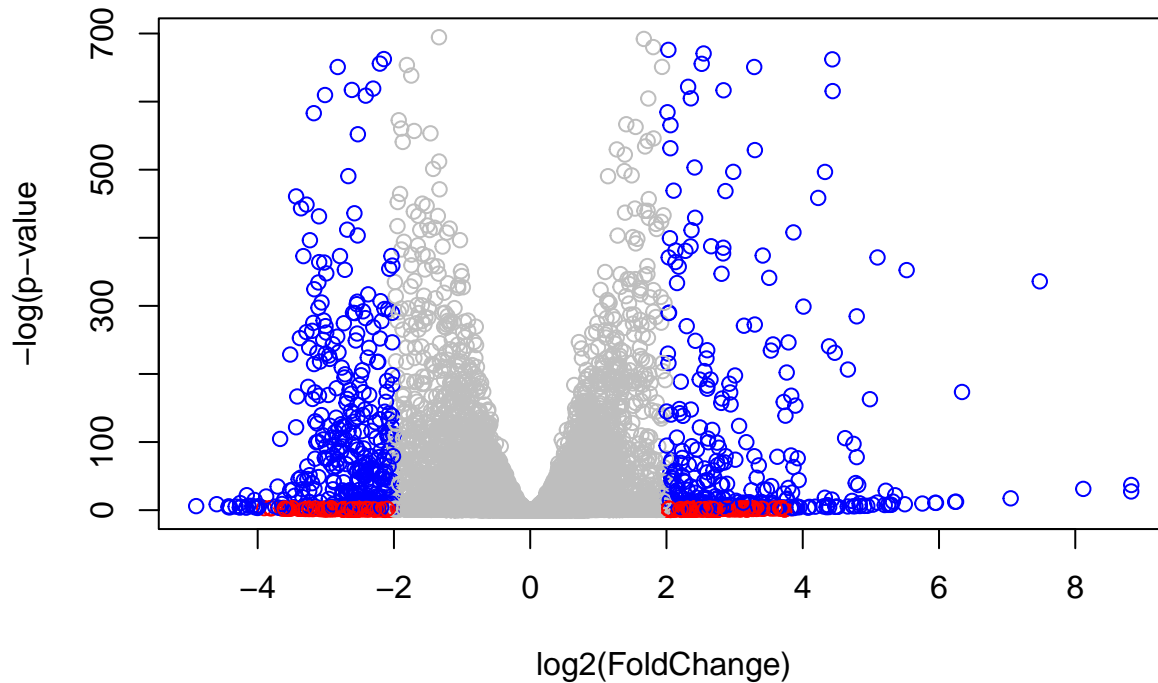
```
plot(res$log2FoldChange, -log(res$padj))
```



Add color

```
mycol <- rep("gray", nrow(res))
mycol[abs(res$log2FoldChange) > 2] <- "blue"
mycol[res$padj > 0.05 & abs(res$log2FoldChange) > 2] <- "red"
```

```
plot(res$log2FoldChange, -log(res$padj), col = mycol, xlab = "log2(FoldChange)",
     ylab = "-log(p-value)")
```



5. Annotation

```
library("AnnotationDbi")
```

```
## Warning: package 'AnnotationDbi' was built under R version 4.1.2
```

```
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    WASH9P    102723897 WAS protein family h..  
## 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 ..
```

6. Pathway Analysis

Use KEGG pathways:

```
library(pathview)  
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"
```

Make the input foldchange vector for KEGG and GO:

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

```
## 102723897 148398 26155 339451 84069 84808
## 0.17925708 0.42645712 -0.69272046 0.72975561 0.04057653 0.54281049
```

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

Look at the object return from gage()

```
attributes(keggres)
```

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

Downregulated pathway: