14. HW-Lab Class14 DESeq2 mini project_rosas

Rosa Chavez (PID: A16487039)

[ ]:
```
library(DESeq2)
```

[2]:
```
metaFile <- "data/GSE37704_metadata.csv"
countFile <- "data/GSE37704_featurecounts.csv"

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

A data.frame: 6 × 1

	condition <chr>
SRR493366	control_sirna
SRR493367	control_sirna
SRR493368	control_sirna
SRR493369	hoxa1_kd
SRR493370	hoxa1_kd
SRR493371	hoxa1_kd

[3]:
```
# Import countdata
countData = read.csv(countFile, row.names=1)
head(countData)
```

A data.frame: 6 × 7

	length <int>	SRR493366 <int>	SRR493367 <int>	SRR493368 <int>	SRR493369 <int>	SRR4 <int>
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

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

[4]:
```
# Note we need to remove the odd first $length col
countData <- as.matrix(countData[,-1])
head(countData)
```

A matrix: 6 × 6 of type int

	SRR493366	SRR493367	SRR493368	SRR493369	SRR493
ENSG00000186092	0	0	0	0	0
ENSG00000279928	0	0	0	0	0
ENSG00000279457	23	28	29	29	28
ENSG00000278566	0	0	0	0	0
ENSG00000273547	0	0	0	0	0
ENSG00000187634	124	123	205	207	212

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).

Tip: What will rowSums() of countData return and how could you use it in this context?

[5]:
```
# Filter count data where you have 0 read count across all samples.
countData = countData[rowSums(countData) > 0, ]
head(countData)
```

A matrix: 6 × 6 of type int

	SRR493366	SRR493367	SRR493368	SRR493369	SRR493
ENSG00000279457	23	28	29	29	28
ENSG00000187634	124	123	205	207	212
ENSG00000188976	1637	1831	2383	1226	1326
ENSG00000187961	120	153	180	236	255
ENSG00000187583	24	48	65	44	48
ENSG00000187642	4	9	16	14	16

[6]:
```
dds = DESeqDataSetFromMatrix(countData=countData,
                             colData=colData,
                             design=~condition)
dds = DESeq(dds)
```

```
Warning message in DESeqDataSet(se, design = design, ignoreRank):
"some variables in design formula are characters, converting to factors"
estimating size factors

estimating dispersions

gene-wise dispersion estimates

mean-dispersion relationship

final dispersion estimates

fitting model and testing
```

[7]:
```
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
```

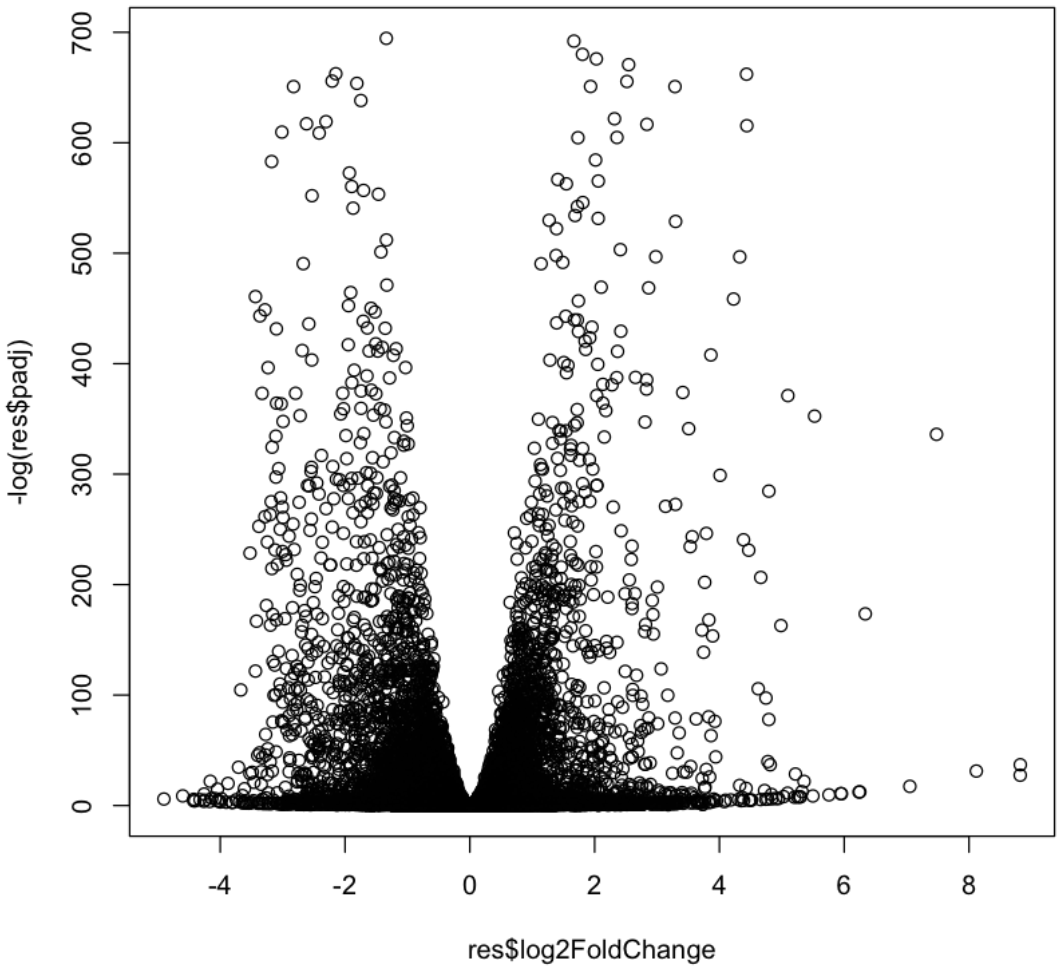
[8]:
```
res = results(dds, contrast=c("condition", "hoxa1_kd", "control_sirna"))
```

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.

[9]:
```
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
```

[10]:
```
plot( res$log2FoldChange, -log(res$padj) )
```



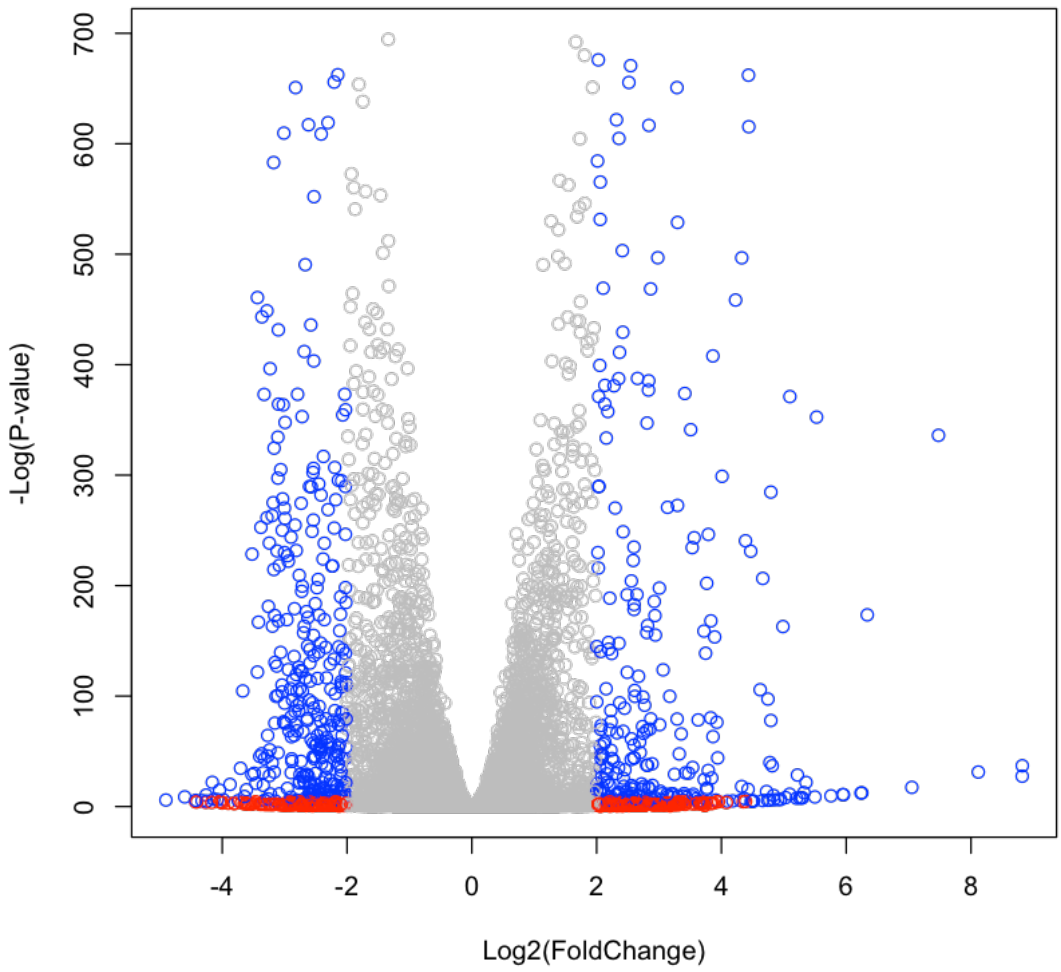
Q. Improve this plot by completing the below code, which adds color and axis labels

[11]:
```
# 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)" )
```



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

[15]:
```
library("AnnotationDbi")
library("org.Hs.eg.db")

columns(org.Hs.eg.db)

res$symbol = mapIds(org.Hs.eg.db,
                    keys=row.names(res),
```

```
                    keytype="ENSEMBL",
                    column="SYMBOL",
                    multiVals="first")

res$entrez = mapIds(org.Hs.eg.db,
                    keys=row.names(res),
                    keytype="ENSEMBL",
                    column="ENTREZID",
                    multiVals="first")

res$name =   mapIds(org.Hs.eg.db,
                    keys=row.names(res),
                    keytype="ENSEMBL",
                    column="ENTREZID",
                    multiVals="first")

head(res, 10)
```

1. 'ACCNUM' 2. 'ALIAS' 3. 'ENSEMBL' 4. 'ENSEMBLPROT' 5. 'ENSEMBLTRANS' 6. 'ENTREZID' 7. 'ENZYME' 8. 'EVIDENCE' 9. 'EVIDENCEALL' 10. 'GENENAME' 11. 'GENETYPE' 12. 'GO' 13. 'GOALL' 14. 'IPI' 15. 'MAP' 16. 'OMIM' 17. 'ONTOLOGY' 18. 'ONTOLOGYALL' 19. 'PATH' 20. 'PFAM' 21. 'PMID' 22. 'PROSITE' 23. 'REFSEQ' 24. 'SYMBOL' 25. 'UCSCKG' 26. 'UNIPROT'

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

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

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


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 1.43989e-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 0.5215599   1.040744 2.97994e-01
ENSG00000188290  108.922128      2.0570638 0.1969053  10.446970 1.51282e-25
ENSG00000187608  350.716868      0.2573837 0.1027266   2.505522 1.22271e-02
ENSG00000188157 9128.439422      0.3899088 0.0467163   8.346304 7.04321e-17
ENSG00000237330    0.158192      0.7859552 4.0804729   0.192614 8.47261e-01
```

```
                     padj      symbol      entrez        name
                <numeric> <character> <character> <character>
ENSG00000279457 6.86555e-01          NA          NA          NA
ENSG00000187634 5.15718e-03      SAMD11      148398      148398
ENSG00000188976 1.76549e-35       NOC2L       26155       26155
ENSG00000187961 1.13413e-07      KLHL17      339451      339451
ENSG00000187583 9.19031e-01     PLEKHN1       84069       84069
ENSG00000187642 4.03379e-01       PERM1       84808       84808
ENSG00000188290 1.30538e-24        HES4       57801       57801
ENSG00000187608 2.37452e-02       ISG15        9636        9636
ENSG00000188157 4.21963e-16        AGRN      375790      375790
ENSG00000237330          NA      RNF223      401934      401934
```

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.

[16]:
```
res <- res[order(res$pvalue),]
write.csv(res, file="deseq.csv")
```

[ ]:
```
# Run in your R console (i.e. not your Rmarkdown doc!)
BiocManager::install( c("pathview", "gage", "gageData") )

# For old vesrsions of R only (R < 3.5.0)!
#source("http://bioconductor.org/biocLite.R")
#biocLite( c("pathview", "gage", "gageData") )
```

[18]:
```
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).
#############################################################################
```

[19]:
```
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' 2. '1544' 3. '1548' 4. '1549' 5. '1553' 6. '7498' 7. '9'

$'hsa00983 Drug metabolism - other enzymes' 1. '10' 2. '1066' 3. '10720' 4. '10941' 5. '151531' 6. '1548' 7. '1549' 8. '1551' 9. '1553' 10. '1576' 11. '1577' 12. '1806' 13. '1807' 14. '1890' 15. '221223' 16. '2990' 17. '3251' 18. '3614' 19. '3615' 20. '3704' 21. '51733' 22. '54490' 23. '54575' 24. '54576' 25. '54577' 26. '54578' 27. '54579' 28. '54600' 29. '54657' 30. '54658' 31. '54659' 32. '54963' 33. '574537' 34. '64816' 35. '7083' 36. '7084' 37. '7172' 38. '7363' 39. '7364' 40. '7365' 41. '7366' 42. '7367' 43. '7371' 44. '7372' 45. '7378' 46. '7498' 47. '79799' 48. '83549' 49. '8824' 50. '8833' 51. '9' 52. '978'

$'hsa00230 Purine metabolism' 1. '100' 2. '10201' 3. '10606' 4. '10621' 5. '10622' 6. '10623' 7. '107' 8. '10714' 9. '108' 10. '10846' 11. '109' 12. '111' 13. '11128' 14. '11164' 15. '112' 16. '113' 17. '114' 18. '115' 19. '122481' 20. '122622' 21. '124583' 22. '132' 23. '158' 24. '159' 25. '1633' 26. '171568' 27. '1716' 28. '196883' 29. '203' 30. '204' 31. '205' 32. '221823' 33. '2272' 34. '22978' 35. '23649' 36. '246721' 37. '25885' 38. '2618' 39. '26289' 40. '270' 41. '271' 42. '27115' 43. '272' 44. '2766' 45. '2977' 46. '2982' 47. '2983' 48. '2984' 49. '2986' 50. '2987' 51. '29922' 52. '3000' 53. '30833' 54. '30834' 55. '318' 56. '3251' 57. '353' 58. '3614' 59. '3615' 60. '3704' 61. '377841' 62. '471' 63. '4830' 64. '4831' 65. '4832' 66. '4833' 67. '4860' 68. '4881' 69. '4882' 70. '4907' 71. '50484' 72. '50940' 73. '51082' 74. '51251' 75. '51292' 76. '5136' 77. '5137' 78. '5138' 79. '5139' 80. '5140' 81. '5141' 82. '5142' 83. '5143' 84. '5144' 85. '5145' 86. '5146' 87. '5147' 88. '5148' 89. '5149' 90. '5150' 91. '5151' 92. '5152' 93. '5153' 94. '5158' 95. '5167' 96. '5169' 97. '51728' 98. '5198' 99. '5236' 100. '5313' 101. '5315' 102. '53343' 103. '54107' 104. '5422' 105. '5424' 106. '5425' 107. '5426' 108. '5427' 109. '5430' 110. '5431' 111. '5432' 112. '5433' 113. '5434' 114. '5435' 115. '5436' 116. '5437' 117. '5438' 118. '5439' 119. '5440' 120. '5441' 121. '5471' 122. '548644' 123. '55276' 124. '5557' 125. '5558' 126. '55703' 127. '55811' 128. '55821' 129. '5631' 130. '5634' 131. '56655' 132. '56953' 133. '56985' 134. '57804' 135. '58497' 136. '6240' 137. '6241' 138. '64425' 139. '646625' 140. '654364' 141. '661' 142. '7498' 143. '8382' 144. '84172' 145. '84265' 146. '84284' 147. '84618' 148. '8622' 149. '8654' 150. '87178' 151. '8833' 152. '9060' 153. '9061' 154. '93034' 155. '953' 156. '9533' 157. '954' 158. '955' 159. '956' 160. '957' 161. '9583' 162. '9615'

[20]:
```
foldchanges = res$log2FoldChange
names(foldchanges) = res$entrez
head(foldchanges)
```

1266 -2.42271923982668 **54855** 3.20195534801527 **1465** -2.31373756504265 **51232** -2.05963137024966 **2034** -1.88801937253936 **2317** -1.6497920067325

[21]:
```
# Get the results
keggres = gage(foldchanges, gsets=kegg.sets.hs)
```

[22]:
```
attributes(keggres)
```

$names = 1. 'greater' 2. 'less' 3. 'stats'

[23]:
```
# Look at the first few down (less) pathways
head(keggres$less)
```

A matrix: 6 × 6 of type dbl

	p.geomean	stat.mean	p.val	q
hsa04110 Cell cycle	8.995727e-06	-4.378644	8.995727e-06	0
hsa03030 DNA replication	9.424076e-05	-3.951803	9.424076e-05	0
hsa03013 RNA transport	1.375901e-03	-3.028500	1.375901e-03	0
hsa03440 Homologous recombination	3.066756e-03	-2.852899	3.066756e-03	0
hsa04114 Oocyte meiosis	3.784520e-03	-2.698128	3.784520e-03	0
hsa00010 Glycolysis / Gluconeogenesis	8.961413e-03	-2.405398	8.961413e-03	0

[24]:
```
pathview(gene.data=foldchanges, pathway.id="hsa04110")
```

```
Info: Downloading xml files for hsa04110, 1/1 pathways..

Info: Downloading png files for hsa04110, 1/1 pathways..

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

Info: Working in directory
/Users/edwinruiz/ComputerScience/BIMM143/late_assignments/lec15

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

[25]:
```
# 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

Warning: reconcile groups sharing member nodes!


     [,1] [,2]
[1,] "9"  "300"
[2,] "9"  "306"

Info: Working in directory
/Users/edwinruiz/ComputerScience/BIMM143/late_assignments/lec15

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

[26]:
```
## 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' 2. 'hsa04630' 3. 'hsa00140' 4. 'hsa04142' 5. 'hsa04330'

[27]:
```
pathview(gene.data=foldchanges, pathway.id=keggresids, species="hsa")
```

```
Info: Downloading xml files for hsa04640, 1/1 pathways..

Info: Downloading png files for hsa04640, 1/1 pathways..

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

Info: Working in directory
/Users/edwinruiz/ComputerScience/BIMM143/late_assignments/lec15

Info: Writing image file hsa04640.pathview.png

Info: Downloading xml files for hsa04630, 1/1 pathways..

Info: Downloading png files for hsa04630, 1/1 pathways..

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

Info: Working in directory
/Users/edwinruiz/ComputerScience/BIMM143/late_assignments/lec15

Info: Writing image file hsa04630.pathview.png

Info: Downloading xml files for hsa00140, 1/1 pathways..

Info: Downloading png files for hsa00140, 1/1 pathways..

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

Info: Working in directory
/Users/edwinruiz/ComputerScience/BIMM143/late_assignments/lec15

Info: Writing image file hsa00140.pathview.png

Info: Downloading xml files for hsa04142, 1/1 pathways..

Info: Downloading png files for hsa04142, 1/1 pathways..

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

Info: Working in directory
/Users/edwinruiz/ComputerScience/BIMM143/late_assignments/lec15

Info: Writing image file hsa04142.pathview.png
```

```
Info: Downloading xml files for hsa04330, 1/1 pathways..

Info: Downloading png files for hsa04330, 1/1 pathways..

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

Info: Working in directory
/Users/edwinruiz/ComputerScience/BIMM143/late_assignments/lec15

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?

[28]:
```
keggrespathways_less <- rownames(keggres$less)[1:5]
keggresids_less <- substr(keggrespathways_less, start=1, stop=8)
pathview(gene.data=foldchanges, pathway.id=keggresids_less, species="hsa")
```

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

Info: Working in directory
/Users/edwinruiz/ComputerScience/BIMM143/late_assignments/lec15

Info: Writing image file hsa04110.pathview.png

Info: Downloading xml files for hsa03030, 1/1 pathways..

Info: Downloading png files for hsa03030, 1/1 pathways..

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

Info: Working in directory
/Users/edwinruiz/ComputerScience/BIMM143/late_assignments/lec15

Info: Writing image file hsa03030.pathview.png

Info: Downloading xml files for hsa03013, 1/1 pathways..

Info: Downloading png files for hsa03013, 1/1 pathways..

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

Info: Working in directory
/Users/edwinruiz/ComputerScience/BIMM143/late_assignments/lec15

Info: Writing image file hsa03013.pathview.png

Info: Downloading xml files for hsa03440, 1/1 pathways..
```

```
Info: Downloading png files for hsa03440, 1/1 pathways..

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

Info: Working in directory
/Users/edwinruiz/ComputerScience/BIMM143/late_assignments/lec15

Info: Writing image file hsa03440.pathview.png

Info: Downloading xml files for hsa04114, 1/1 pathways..

Info: Downloading png files for hsa04114, 1/1 pathways..

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

Info: Working in directory
/Users/edwinruiz/ComputerScience/BIMM143/late_assignments/lec15

Info: Writing image file hsa04114.pathview.png
```

[29]:
```
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 A matrix: 6 × 6 of type dbl

	p.geomean	stat.mean
GO:0007156 homophilic cell adhesion	8.519724e-05	3.824205
GO:0002009 morphogenesis of an epithelium	1.396681e-04	3.653886
GO:0048729 tissue morphogenesis	1.432451e-04	3.643242
GO:0007610 behavior	1.925222e-04	3.565432
GO:0060562 epithelial tube morphogenesis	5.932837e-04	3.261376
GO:0035295 tube development	5.953254e-04	3.253665

$less A matrix: 6 × 6 of type dbl

	p.geomean	stat.mean	p.val
GO:0048285 organelle fission	1.536227e-15	-8.063910	1.53622
GO:0000280 nuclear division	4.286961e-15	-7.939217	4.28696
GO:0007067 mitosis	4.286961e-15	-7.939217	4.28696
GO:0000087 M phase of mitotic cell cycle	1.169934e-14	-7.797496	1.16993
GO:0007059 chromosome segregation	2.028624e-11	-6.878340	2.02862
GO:0000236 mitotic prometaphase	1.729553e-10	-6.695966	1.72955

$stats A matrix: 6 × 2 of type dbl

	stat.mean	exp1
GO:0007156 homophilic cell adhesion	3.824205	3.824205
GO:0002009 morphogenesis of an epithelium	3.653886	3.653886
GO:0048729 tissue morphogenesis	3.643242	3.643242
GO:0007610 behavior	3.565432	3.565432
GO:0060562 epithelial tube morphogenesis	3.261376	3.261376
GO:0035295 tube development	3.253665	3.253665

[30]:
```
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: 8147"
```

[31]:
```
write.table(sig_genes, file="significant_genes.txt", row.names=FALSE, col.
  ↪names=FALSE, quote=FALSE)
```

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 Cell Cycle has the most significant "Entities p-value" 2.43E-4. Yes, the most significant pathways listed match the KEGG results "hsa04110 Cell cycle". The factors that can cause the difference between these two methods can be the gene identifiers or statistical approaches.