

# class14

Jaewon Kim

```
#|eval: FALSE  
library(DESeq2)
```

```
: S4Vectors
```

```
: stats4
```

```
: BiocGenerics
```

```
: 'BiocGenerics'
```

The following objects are masked from 'package:stats':

IQR, mad, sd, var, xtabs

The following objects are masked from 'package:base':

anyDuplicated, aperm, 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

```
: 'S4Vectors'
```

The following object is masked from 'package:utils':

findMatches

The following objects are masked from 'package:base':

expand.grid, I, unname

: IRanges

: 'IRanges'

The following object is masked from 'package:grDevices':

windows

: GenomicRanges

: GenomeInfoDb

: SummarizedExperiment

: MatrixGenerics

: matrixStats

: 'MatrixGenerics'

The following objects are masked from 'package:matrixStats':

colAlls, colAnyNAs, colAnys, colAvgPerRowSet, 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
```

```
: Biobase
```

Welcome to Bioconductor

```
Vignettes contain introductory material; view with
'browseVignettes()'. To cite Bioconductor, see
'citation("Biobase")', and for packages 'citation("pkgname")'.
```

```
: '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"

colData = read.csv(metafile, row.names=1)
countData = read.csv(countfile, row.names=1)
head(colData)
```

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

```
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

```
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

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).

```
zero.ind <- rowSums((countData[, 1:length(countData[1,])] == 0)) > 0
countData <- countData[!zero.ind, ]
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

```
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: 13282 6
metadata(1): version
assays(4): counts mu H cooks
rownames(13282): ENSG00000279457 ENSG00000187634 ... ENSG00000276345
               ENSG00000271254
rowData names(22): baseMean baseVar ... deviance maxCooks
colnames(6): SRR493366 SRR493367 ... SRR493370 SRR493371
colData names(2): condition sizeFactor
```

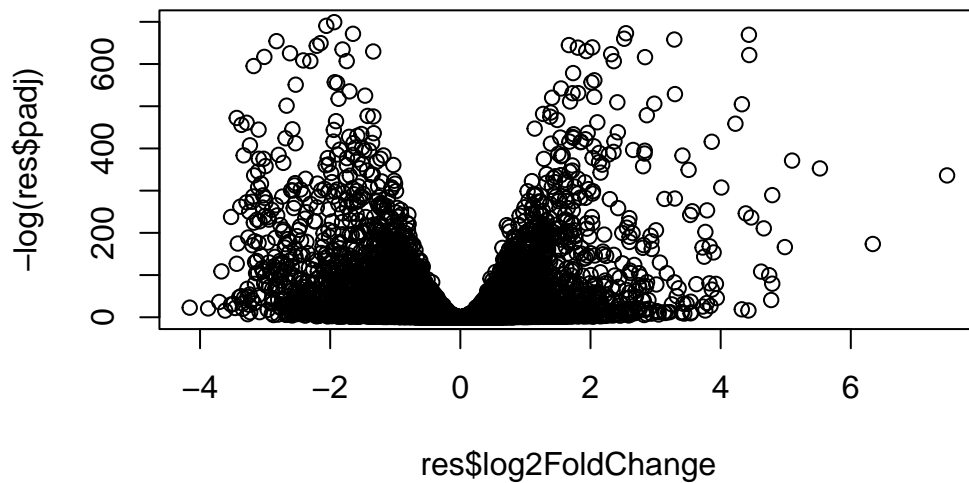
```
res = results(dds, contrast=c("condition", "hoxa1_kd", "control_sirna"))
```

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 13282 with nonzero total read count
adjusted p-value < 0.1
LFC > 0 (up)      : 4333, 33%
LFC < 0 (down)    : 4400, 33%
outliers [1]      : 0, 0%
low counts [2]     : 0, 0%
(mean count < 1)
[1] see 'cooksCutoff' argument of ?results
[2] see 'independentFiltering' argument of ?results
```

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



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

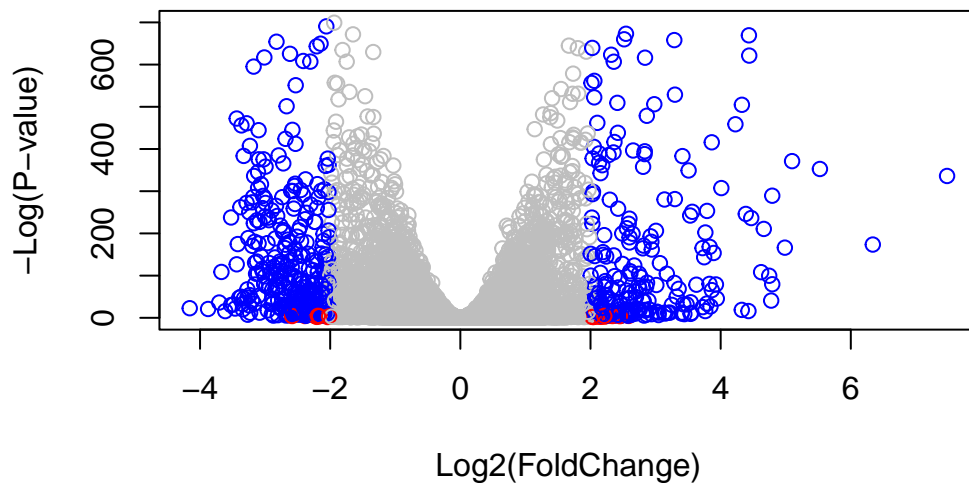
```

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

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

# Color blue those with adjusted p-value less than 0.01
# and absolute fold change more than 2
inds <- (res$padj < 0.01) & (standard)
mycols[inds] <- "blue"
plot( res$log2FoldChange, -log(res$padj), col= mycols, xlab="Log2(FoldChange)", ylab="-Log"

```



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$genename = 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	0.1803039	0.3121566	0.577607	5.63529e-01
ENSG00000187634	183.2296	0.4258966	0.1355303	3.142446	1.67543e-03
ENSG00000188976	1651.1881	-0.6927118	0.0549876	-12.597612	2.17635e-36
ENSG00000187961	209.6379	0.7299597	0.1277613	5.713463	1.10700e-08
ENSG00000187583	47.2551	0.0392549	0.2606192	0.150622	8.80274e-01
ENSG00000187642	11.9798	0.5395082	0.5001355	1.078724	2.80711e-01
ENSG00000188290	108.9221	2.0562855	0.1910714	10.761870	5.21018e-27
ENSG00000187608	350.7169	0.2570251	0.0999769	2.570845	1.01451e-02
ENSG00000188157	9128.4394	0.3899096	0.0482214	8.085827	6.17439e-16
ENSG00000131591	156.4791	0.1968918	0.1406800	1.399572	1.61641e-01

	padj	symbol	entrez	genename
	<numeric>	<character>	<character>	<character>
ENSG00000279457	6.47026e-01	NA	NA	NA
ENSG00000187634	3.34029e-03	SAMD11	148398	sterile alpha motif ..
ENSG00000188976	2.35970e-35	NOC2L	26155	NOC2 like nucleolar ..
ENSG00000187961	3.69612e-08	KLHL17	339451	kelch like family me..
ENSG00000187583	9.10931e-01	PLEKHN1	84069	pleckstrin homology ..
ENSG00000187642	3.61174e-01	PERM1	84808	PPARGC1 and ESRR ind..
ENSG00000188290	4.17884e-26	HES4	57801	hes family bHLH tran..
ENSG00000187608	1.79950e-02	ISG15	9636	ISG15 ubiquitin like..
ENSG00000188157	3.15902e-15	AGRN	375790	agrin
ENSG00000131591	2.23894e-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.

```
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	-2.42268	0.0612493	-39.5545	0

ENSG00000183508	2053.88	3.20186	0.0728530	43.9496	0
ENSG00000159176	5692.46	-2.31371	0.0589160	-39.2714	0
ENSG00000116016	4423.95	-1.88800	0.0442896	-42.6285	0
ENSG00000164251	2348.77	3.34448	0.0695586	48.0815	0
ENSG00000124766	2576.65	2.39226	0.0623261	38.3829	0
	padj	symbol	entrez		genename
	<numeric>	<character>	<character>		<character>
ENSG00000117519	0	CNN3	1266		calponin 3
ENSG00000183508	0	TENT5C	54855	terminal nucleotidyl..	
ENSG00000159176	0	CSRP1	1465	cysteine and glycine..	
ENSG00000116016	0	EPAS1	2034	endothelial PAS doma..	
ENSG00000164251	0	F2RL1	2150	F2R like trypsin rec..	
ENSG00000124766	0	SOX4	6659	SRY-box transcriptio..	

```
write.csv(res[ord,], "deseq_results.csv")
```

```
# BiocManager::install(c("pathview", "gage", "gageData"))
```

```
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)
```

```
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"
```

```
$`hsa01100 Metabolic pathways`
```

```
[1] "10" "100" "10007" "100137049" "10020" "10026"  
[7] "100510686" "10063" "10157" "10170" "10195" "10201"  
[13] "10229" "10312" "10317" "10327" "10331" "1036"  
[19] "10380" "10390" "1040" "10400" "10402" "10423"  
[25] "10449" "10476" "10554" "10555" "10558" "1056"  
[31] "10588" "10606" "10621" "10622" "10623" "10632"  
[37] "10654" "1066" "10678" "10682" "10690" "10714"  
[43] "10720" "10768" "10797" "10826" "10841" "10855"  
[49] "10873" "10901" "10905" "10941" "10975" "10993"  
[55] "10998" "11019" "11041" "1109" "11112" "11128"  
[61] "1119" "1120" "11226" "11227" "11232" "112483"  
[67] "11253" "11282" "11285" "113026" "11320" "11343"  
[73] "113451" "113612" "114805" "1152" "1158" "1159"  
[79] "1160" "116285" "117248" "119548" "120227" "121278"  
[85] "122481" "122622" "123099" "123745" "123876" "124"  
[91] "124454" "124975" "125" "125061" "125965" "125981"  
[97] "126" "126328" "126792" "127" "127124" "128"  
[103] "128869" "129607" "129642" "130" "130013" "131"  
[109] "1312" "131669" "132" "132158" "1327" "132789"  
[115] "1329" "1337" "1339" "1340" "134147" "1345"  
[121] "1349" "1350" "1351" "135152" "1352" "1353"  
[127] "1355" "1371" "1373" "137964" "138050" "138429"  
[133] "139596" "140838" "1431" "144193" "144245" "145226"  
[139] "146664" "1491" "15" "1503" "150763" "151056"  
[145] "151531" "1537" "154141" "1543" "1544" "1548"  
[151] "1549" "155066" "1551" "1553" "1555" "1557"
```

[157]	"1558"	"1559"	"1562"	"1571"	"1573"	"157506"
[163]	"1576"	"1577"	"1579"	"158"	"1581"	"1582"
[169]	"1583"	"1584"	"1585"	"1586"	"1588"	"1589"
[175]	"159"	"1593"	"1594"	"1595"	"160287"	"1603"
[181]	"1606"	"1607"	"1608"	"160851"	"1609"	"1610"
[187]	"1621"	"162417"	"162466"	"1629"	"1633"	"1635"
[193]	"1638"	"1644"	"1650"	"166929"	"168391"	"169355"
[199]	"170712"	"171568"	"1716"	"1717"	"1718"	"1719"
[205]	"1723"	"1737"	"1738"	"1743"	"1757"	"178"
[211]	"1786"	"1787"	"1788"	"1789"	"1798"	"18"
[217]	"1806"	"1807"	"1841"	"1854"	"189"	"1890"
[223]	"1892"	"191"	"192134"	"1962"	"197258"	"199857"
[229]	"201595"	"2023"	"2026"	"2027"	"203"	"204"
[235]	"205"	"2053"	"2058"	"210"	"211"	"212"
[241]	"2131"	"2132"	"2134"	"2135"	"2137"	"216"
[247]	"217"	"218"	"2180"	"2181"	"2182"	"2184"
[253]	"219"	"2194"	"220"	"2203"	"221"	"221223"
[259]	"221823"	"222"	"2222"	"2224"	"223"	"2235"
[265]	"224"	"226"	"2271"	"22845"	"22856"	"229"
[271]	"22928"	"22929"	"22934"	"22978"	"230"	"23057"
[277]	"231"	"23193"	"23236"	"23305"	"23382"	"23396"
[283]	"23417"	"23475"	"23483"	"23498"	"23530"	"23545"
[289]	"23553"	"23556"	"2356"	"23600"	"23649"	"23761"
[295]	"239"	"240"	"242"	"245972"	"245973"	"246"
[301]	"246721"	"247"	"248"	"249"	"250"	"251"
[307]	"2523"	"2524"	"2525"	"2526"	"2527"	"2528"
[313]	"2529"	"2530"	"2531"	"253558"	"2538"	"2539"
[319]	"254531"	"2548"	"256435"	"2571"	"2572"	"25796"
[325]	"2581"	"2582"	"2583"	"25834"	"2584"	"2585"
[331]	"2588"	"25885"	"2589"	"2590"	"25902"	"2591"
[337]	"2592"	"259230"	"2593"	"259307"	"2595"	"2597"
[343]	"26007"	"26035"	"2618"	"262"	"26227"	"26229"
[349]	"26275"	"26279"	"2628"	"26289"	"2629"	"26290"
[355]	"26301"	"2632"	"26330"	"2639"	"2643"	"2645"
[361]	"2650"	"2651"	"2673"	"2678"	"2683"	"2686"
[367]	"2687"	"270"	"27010"	"27034"	"27087"	"27089"
[373]	"27090"	"271"	"2710"	"2712"	"27124"	"27165"
[379]	"272"	"2720"	"27235"	"2729"	"2730"	"27306"
[385]	"2731"	"27349"	"27430"	"2744"	"2746"	"2747"
[391]	"275"	"2752"	"276"	"2762"	"277"	"278"
[397]	"279"	"2799"	"28"	"280"	"2805"	"2806"
[403]	"2821"	"283208"	"283871"	"284098"	"284541"	"2875"
[409]	"290"	"29071"	"2937"	"2954"	"29796"	"2987"

[415]	"29880"	"2990"	"29906"	"29920"	"29922"	"29925"
[421]	"29926"	"29929"	"29947"	"29958"	"29968"	"30"
[427]	"3028"	"3030"	"3032"	"3033"	"3034"	"3067"
[433]	"3073"	"3074"	"3081"	"30814"	"30815"	"30833"
[439]	"30834"	"3098"	"3099"	"31"	"3101"	"314"
[445]	"3141"	"3145"	"3155"	"3156"	"3157"	"3158"
[451]	"316"	"317749"	"32"	"3242"	"3251"	"326625"
[457]	"3283"	"3284"	"3290"	"3291"	"3292"	"3293"
[463]	"3294"	"3295"	"33"	"3340"	"3373"	"337876"
[469]	"339221"	"34"	"340485"	"341392"	"3417"	"3418"
[475]	"3419"	"341947"	"3420"	"3421"	"3422"	"3423"
[481]	"3425"	"348158"	"349565"	"35"	"353"	"36"
[487]	"3612"	"3613"	"3614"	"3615"	"3620"	"3628"
[493]	"3631"	"3632"	"3633"	"3636"	"37"	"3703"
[499]	"3704"	"3705"	"3706"	"3707"	"3712"	"374291"
[505]	"374378"	"3795"	"38"	"383"	"384"	"387787"
[511]	"39"	"3906"	"391013"	"3938"	"3939"	"3945"
[517]	"3948"	"3990"	"4047"	"4048"	"4051"	"4056"
[523]	"411"	"4121"	"4122"	"4124"	"4128"	"4129"
[529]	"4143"	"4144"	"4190"	"4191"	"4199"	"4245"
[535]	"4247"	"4248"	"4249"	"427"	"4329"	"435"
[541]	"4351"	"4357"	"438"	"440"	"440138"	"440567"
[547]	"441024"	"441531"	"442117"	"445"	"4507"	"4508"
[553]	"4509"	"4512"	"4513"	"4514"	"4519"	"4522"
[559]	"4524"	"4535"	"4536"	"4537"	"4538"	"4539"
[565]	"4540"	"4541"	"4548"	"4594"	"4597"	"4598"
[571]	"4669"	"4694"	"4695"	"4696"	"4697"	"4698"
[577]	"47"	"4700"	"4701"	"4702"	"4704"	"4705"
[583]	"4706"	"4707"	"4708"	"4709"	"471"	"4710"
[589]	"4711"	"4712"	"4713"	"4714"	"4715"	"4716"
[595]	"4717"	"4718"	"4719"	"4720"	"4722"	"4723"
[601]	"4724"	"4725"	"4726"	"4728"	"4729"	"4731"
[607]	"48"	"4830"	"4831"	"4832"	"4833"	"4837"
[613]	"4842"	"4843"	"4846"	"4860"	"4907"	"493911"
[619]	"4942"	"4952"	"4953"	"4967"	"498"	"50"
[625]	"5009"	"501"	"5033"	"5048"	"50484"	"50487"
[631]	"5049"	"5050"	"5051"	"5053"	"506"	"50614"
[637]	"50617"	"50700"	"50814"	"509"	"5091"	"5095"
[643]	"5096"	"51"	"51004"	"5105"	"51056"	"5106"
[649]	"51074"	"51082"	"51084"	"51102"	"51109"	"51144"
[655]	"51166"	"51179"	"51181"	"51196"	"51227"	"51251"
[661]	"51268"	"513"	"5130"	"51301"	"51380"	"51382"
[667]	"514"	"51477"	"51478"	"515"	"51540"	"516"

[673]	"5160"	"51601"	"51604"	"51606"	"5161"	"5162"
[679]	"5167"	"5169"	"517"	"51703"	"51727"	"51728"
[685]	"51733"	"51763"	"518"	"51805"	"51809"	"5198"
[691]	"521"	"5211"	"5213"	"5214"	"522"	"5223"
[697]	"5224"	"5226"	"523"	"5230"	"5232"	"5236"
[703]	"525"	"526"	"527"	"5277"	"5279"	"528"
[709]	"5281"	"5283"	"5286"	"5287"	"5288"	"5289"
[715]	"529"	"5297"	"5298"	"5313"	"5315"	"5319"
[721]	"5320"	"5321"	"5322"	"533"	"5330"	"5331"
[727]	"5332"	"5333"	"5335"	"53354"	"5336"	"5337"
[733]	"5338"	"534"	"535"	"53630"	"537"	"5372"
[739]	"5373"	"539"	"53947"	"5406"	"5407"	"5408"
[745]	"5409"	"54107"	"54187"	"5422"	"5424"	"5425"
[751]	"5426"	"5427"	"5428"	"5430"	"5431"	"5432"
[757]	"5433"	"5434"	"54344"	"5435"	"5436"	"54363"
[763]	"5437"	"5438"	"5439"	"5440"	"5441"	"5444"
[769]	"5445"	"5446"	"54480"	"54490"	"54575"	"54576"
[775]	"54577"	"54578"	"54579"	"54600"	"54657"	"54658"
[781]	"54659"	"54675"	"5471"	"54802"	"548596"	"548644"
[787]	"549"	"54947"	"54963"	"54965"	"5498"	"54988"
[793]	"54995"	"55163"	"55191"	"55224"	"55229"	"55256"
[799]	"55276"	"55300"	"55301"	"55304"	"55312"	"55361"
[805]	"5538"	"55454"	"55500"	"55512"	"55568"	"5557"
[811]	"5558"	"55627"	"55650"	"55703"	"55750"	"55753"
[817]	"55790"	"55808"	"55821"	"55902"	"55907"	"56052"
[823]	"5625"	"56267"	"5631"	"5634"	"56474"	"56623"
[829]	"56624"	"56655"	"56848"	"56894"	"56895"	"56898"
[835]	"56901"	"56913"	"56922"	"56953"	"56994"	"570"
[841]	"57016"	"57026"	"57134"	"5723"	"5730"	"5740"
[847]	"5742"	"5743"	"57452"	"574537"	"57678"	"57804"
[853]	"57818"	"57834"	"5805"	"5831"	"5832"	"5833"
[859]	"58510"	"5859"	"586"	"5860"	"587"	"593"
[865]	"594"	"5980"	"60490"	"60495"	"6120"	"6184"
[871]	"6185"	"622"	"6240"	"6241"	"6296"	"6303"
[877]	"6307"	"6309"	"6342"	"635"	"6389"	"6390"
[883]	"6391"	"63917"	"6392"	"64087"	"64131"	"64132"
[889]	"64409"	"64425"	"6448"	"64579"	"64600"	"646625"
[895]	"6470"	"6472"	"6476"	"64768"	"6480"	"64802"
[901]	"64816"	"6482"	"6483"	"6484"	"6487"	"6489"
[907]	"64902"	"65220"	"65263"	"654364"	"6609"	"661"
[913]	"6610"	"6611"	"6652"	"6675"	"6677"	"669"
[919]	"6697"	"6713"	"6718"	"6723"	"683"	"686"
[925]	"6888"	"6898"	"6916"	"6999"	"7054"	"7083"

[931]	"7084"	"7086"	"7108"	"7166"	"7167"	"7173"
[937]	"7263"	"7264"	"729020"	"7298"	"7299"	"7306"
[943]	"7357"	"7358"	"7360"	"7363"	"7364"	"7365"
[949]	"7366"	"7367"	"7368"	"7371"	"7372"	"7378"
[955]	"7381"	"7384"	"7385"	"7386"	"7388"	"7389"
[961]	"7390"	"7498"	"7841"	"790"	"79053"	"79087"
[967]	"7915"	"79178"	"7923"	"79369"	"7941"	"79586"
[973]	"79611"	"79623"	"79646"	"79695"	"79717"	"79796"
[979]	"79799"	"79814"	"79868"	"79888"	"7991"	"80025"
[985]	"80055"	"80142"	"80146"	"80201"	"80270"	"80308"
[991]	"80339"	"80347"	"8050"	"81490"	"81579"	"81616"
[997]	"81849"	"81888"	"8277"	"8309"	"8310"	"83440"
[1003]	"83549"	"8372"	"8382"	"8394"	"8395"	"8398"
[1009]	"8399"	"84002"	"84076"	"84172"	"84245"	"84265"
[1015]	"84274"	"84284"	"84532"	"84618"	"84620"	"84647"
[1021]	"84649"	"84693"	"847"	"84701"	"84706"	"84720"
[1027]	"84735"	"84803"	"84812"	"84890"	"84920"	"84992"
[1033]	"8509"	"8513"	"8525"	"8526"	"8527"	"8529"
[1039]	"85365"	"8540"	"85465"	"8560"	"8564"	"8566"
[1045]	"8608"	"8611"	"8612"	"8613"	"8630"	"8639"
[1051]	"8659"	"8681"	"8692"	"8693"	"8694"	"8702"
[1057]	"8703"	"8704"	"8705"	"8706"	"8707"	"8708"
[1063]	"873"	"8733"	"874"	"875"	"8760"	"8789"
[1069]	"8790"	"8801"	"8802"	"8803"	"8813"	"8818"
[1075]	"8821"	"883"	"8833"	"8854"	"8867"	"8869"
[1081]	"8871"	"8877"	"8879"	"8942"	"8972"	"8974"
[1087]	"89869"	"8992"	"9"	"90423"	"9060"	"9061"
[1093]	"9091"	"9114"	"91373"	"9162"	"91734"	"9197"
[1099]	"9245"	"92483"	"9249"	"9296"	"93034"	"93183"
[1105]	"9331"	"9348"	"9374"	"9377"	"9380"	"9388"
[1111]	"94005"	"9453"	"9468"	"9487"	"9488"	"9489"
[1117]	"95"	"9514"	"9517"	"952"	"9533"	"9536"
[1123]	"9550"	"9551"	"9563"	"9588"	"9615"	"978"
[1129]	"9791"	"9942"	"9945"			

```

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

```

<NA>	148398	26155	339451	84069	84808
0.18030389	0.42589661	-0.69271181	0.72995968	0.03925487	0.53950816

```
keggres = gage(foldchanges, gsets=kegg.sets.hs)
```

```
attributes(keggres)
```

```
$names
```

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

```
head(keggres$less)
```

	p.geomean	stat.mean
hsa04110 Cell cycle	3.548176e-06	-4.604234
hsa03030 DNA replication	3.992330e-05	-4.191094
hsa04114 Oocyte meiosis	2.332810e-04	-3.564509
hsa05130 Pathogenic Escherichia coli infection	6.474555e-04	-3.326950
hsa03440 Homologous recombination	2.248158e-03	-2.967340
hsa03013 RNA transport	3.743589e-03	-2.698970

	p.val	q.val
hsa04110 Cell cycle	3.548176e-06	0.0007202798
hsa03030 DNA replication	3.992330e-05	0.0040522154
hsa04114 Oocyte meiosis	2.332810e-04	0.0157853509
hsa05130 Pathogenic Escherichia coli infection	6.474555e-04	0.0328583656
hsa03440 Homologous recombination	2.248158e-03	0.0912752208
hsa03013 RNA transport	3.743589e-03	0.1266580962

	set.size	exp1
hsa04110 Cell cycle	118	3.548176e-06
hsa03030 DNA replication	36	3.992330e-05
hsa04114 Oocyte meiosis	95	2.332810e-04
hsa05130 Pathogenic Escherichia coli infection	46	6.474555e-04
hsa03440 Homologous recombination	28	2.248158e-03
hsa03013 RNA transport	140	3.743589e-03

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

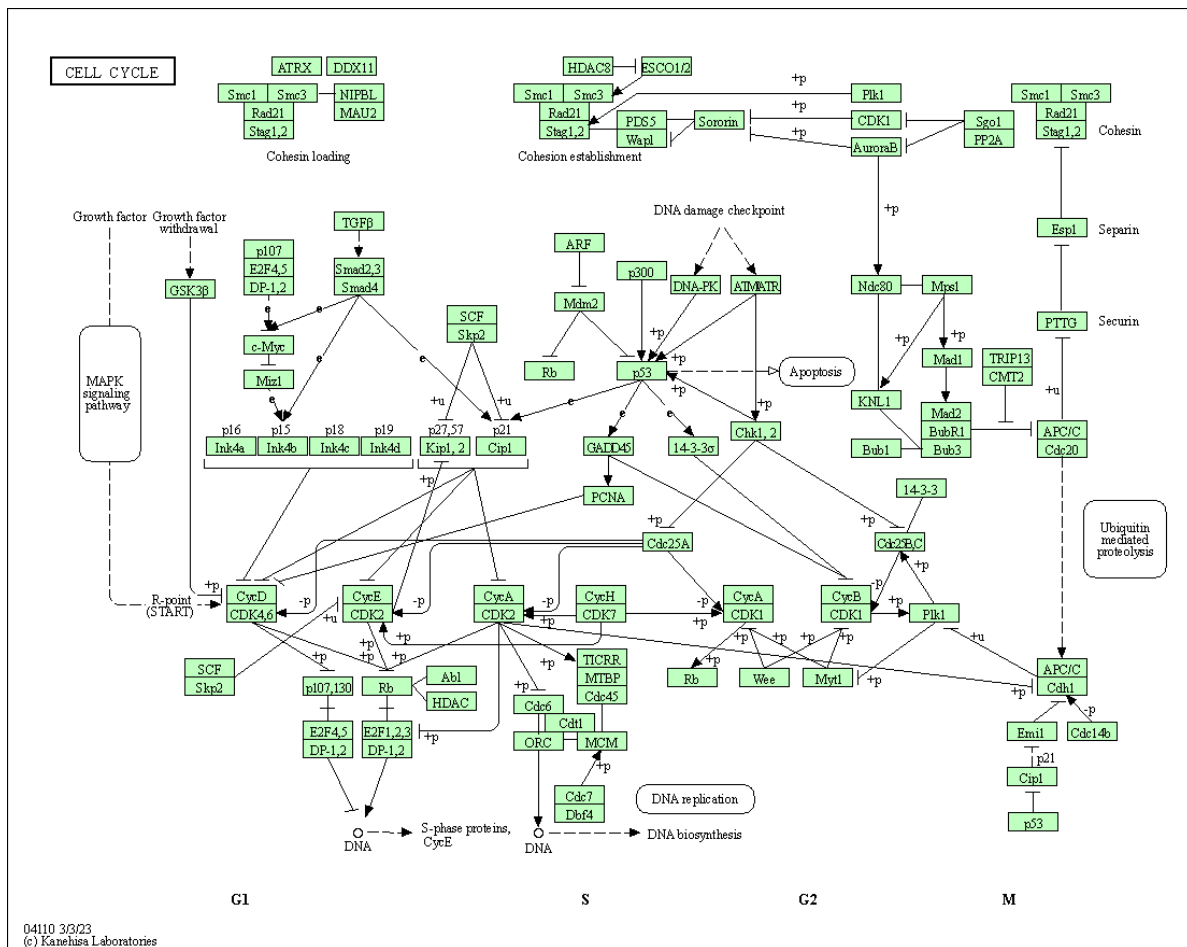
```
'select()' returned 1:1 mapping between keys and columns
```

```
Info: Working in directory C:/Users/louis/Downloads/bimm143/R code/class14/class14
```

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







## 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] "hsa04142" "hsa05323" "hsa04060" "hsa04640" "hsa05332"
```

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

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

Info: Working in directory C:/Users/louis/Downloads/bimm143/R code/class14/class14

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 C:/Users/louis/Downloads/bimm143/R code/class14/class14

Info: Writing image file hsa05323.pathview.png

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

Info: Working in directory C:/Users/louis/Downloads/bimm143/R code/class14/class14

Info: Writing image file hsa04060.pathview.png

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

Info: Working in directory C:/Users/louis/Downloads/bimm143/R code/class14/class14

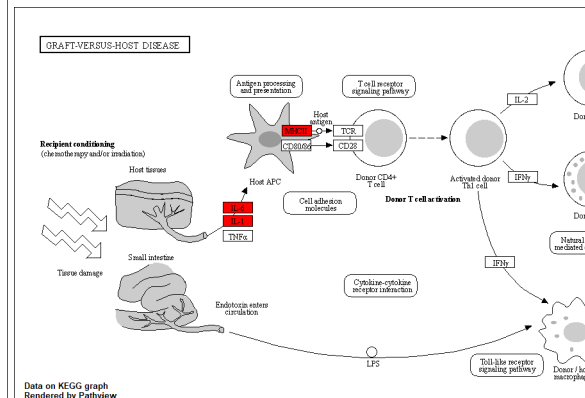
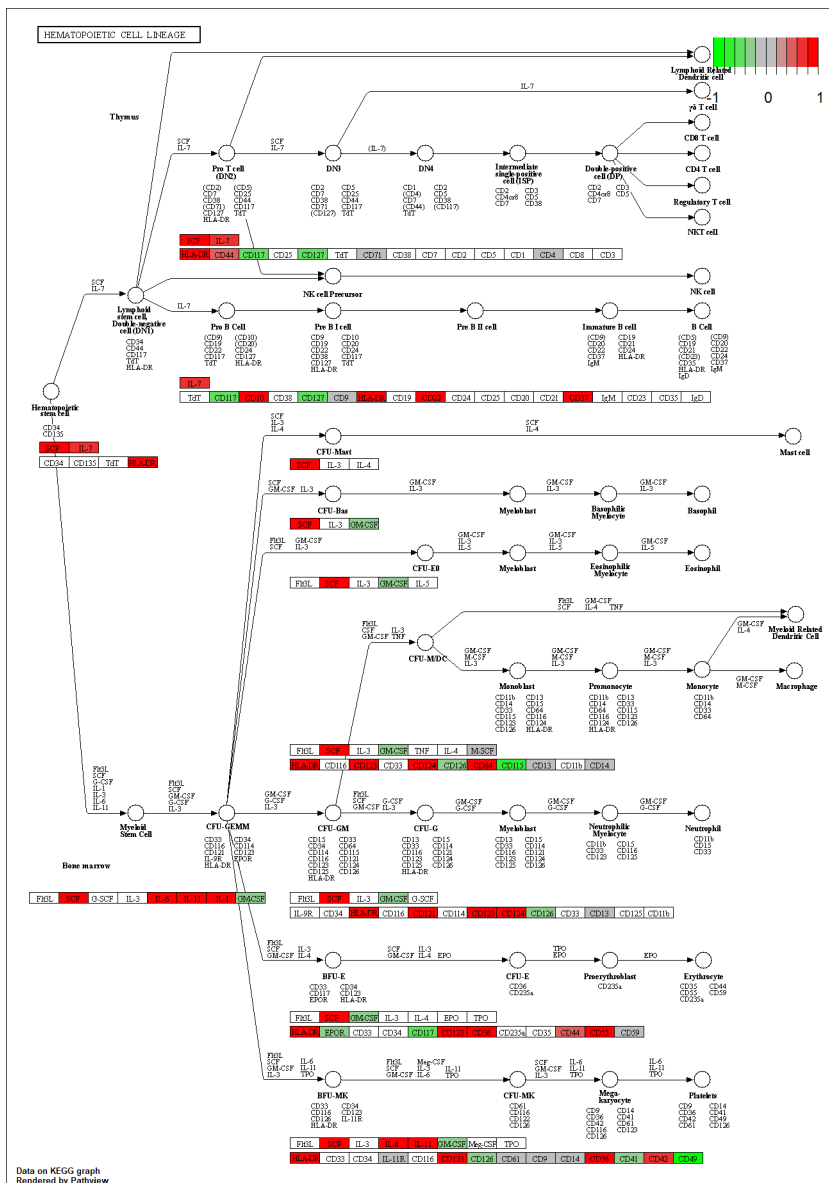
Info: Writing image file hsa04640.pathview.png

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

Info: Working in directory C:/Users/louis/Downloads/bimm143/R code/class14/class14

Info: Writing image file hsa05332.pathview.png





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

```
questionpathways <- rownames(keggres$less)[1:5]

questioninput = substr(questionpathways, start=1, stop=8)
questioninput
```

[1] "hsa04110" "hsa03030" "hsa04114" "hsa05130" "hsa03440"

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

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

Info: Working in directory C:/Users/louis/Downloads/bimm143/R code/class14/class14

Info: Writing image file hsa04110.pathview.png

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

Info: Working in directory C:/Users/louis/Downloads/bimm143/R code/class14/class14

Info: Writing image file hsa03030.pathview.png

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

Info: Working in directory C:/Users/louis/Downloads/bimm143/R code/class14/class14

Info: Writing image file hsa04114.pathview.png

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

Info: Working in directory C:/Users/louis/Downloads/bimm143/R code/class14/class14

Info: Writing image file hsa05130.pathview.png

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

Info: Working in directory C:/Users/louis/Downloads/bimm143/R code/class14/class14

Info: Writing image file hsa03440.pathview.png









G0:0008285	negative regulation of cell proliferation	1.443571e-03	0.5718590
G0:0051047	positive regulation of secretion	1.877703e-03	0.5718590
		set.size	exp1
G0:0007156	homophilic cell adhesion	90	7.523307e-05
G0:0016339	calcium-dependent cell-cell adhesion	24	8.556504e-04
G0:0010817	regulation of hormone levels	225	1.058523e-03
G0:0048729	tissue morphogenesis	347	1.389102e-03
G0:0008285	negative regulation of cell proliferation	386	1.443571e-03
G0:0051047	positive regulation of secretion	130	1.877703e-03

\$less

		p.geomean	stat.mean	p.val
G0:0000279	M phase	6.451975e-18	-8.738701	6.451975e-18
G0:0048285	organelle fission	1.832907e-16	-8.369971	1.832907e-16
G0:0000280	nuclear division	2.627088e-16	-8.340038	2.627088e-16
G0:0007067	mitosis	2.627088e-16	-8.340038	2.627088e-16
G0:0000087	M phase of mitotic cell cycle	9.244549e-16	-8.166584	9.244549e-16
G0:0007059	chromosome segregation	2.502912e-12	-7.264756	2.502912e-12
		q.val	set.size	exp1
G0:0000279	M phase	2.398199e-14	467	6.451975e-18
G0:0048285	organelle fission	2.441221e-13	360	1.832907e-16
G0:0000280	nuclear division	2.441221e-13	338	2.627088e-16
G0:0007067	mitosis	2.441221e-13	338	2.627088e-16
G0:0000087	M phase of mitotic cell cycle	6.872398e-13	348	9.244549e-16
G0:0007059	chromosome segregation	1.550554e-09	135	2.502912e-12

\$stats

		stat.mean	exp1
G0:0007156	homophilic cell adhesion	3.873939	3.873939
G0:0016339	calcium-dependent cell-cell adhesion	3.340855	3.340855
G0:0010817	regulation of hormone levels	3.091986	3.091986
G0:0048729	tissue morphogenesis	3.002504	3.002504
G0:0008285	negative regulation of cell proliferation	2.989717	2.989717
G0:0051047	positive regulation of secretion	2.927781	2.927781

```
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: 8186"
```

```
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?

On KEGG, cell cycle had lowest p-value, where GO result had ‘biological process’ (p-value  $8.25e-74$ ) of stimulus detection pathway as lowest. The difference between KEGG and GO is that KEGG computes gene length and abundance bias through tricubemovingaverage() where as GO uses monotonic regression.