Class 14: RNA-Seq analysis mini-project

Georgina Canto-Encalada

Section 1. Differential Expression Analysis

Data import

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

# Import metadata and take a peak
metadata<- read.csv(metaFile)
counts<- read.csv(countFile, row.names=1)
```

Data exploration

```
head(counts)
```

```
                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
```

```
head(metadata)
```

```
        id     condition
1 SRR493366 control_sirna
2 SRR493367 control_sirna
3 SRR493368 control_sirna
4 SRR493369      hoxa1_kd
5 SRR493370      hoxa1_kd
6 SRR493371      hoxa1_kd
```

Check for metadata - counts correspond

```
countdata<-counts[,-1]
all(colnames(countdata)==metadata$id)
```

```
[1] TRUE
```

Filter out zero count genes

We can sum across the rows and then remove those with zero sums

```
non.zero.inds <- rowSums(countdata)>0
non.zero.counts<-countdata[non.zero.inds,]
head(non.zero.counts)
```

```
                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
```

Q. How many genes do we have with non zero counts?

```
nrow(non.zero.counts)
```

```
[1] 15975
```

Setup for Deseq

```
#/ message: false
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, 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
```

```

Attaching package: 'S4Vectors'
```

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

    findMatches
```

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

    expand.grid, I, unname
```

```
Loading required package: IRanges
```

```
Attaching package: 'IRanges'

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

    windows

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
```

DESeq Analysis

```
dds<-DESeqDataSetFromMatrix(countData=non.zero.counts,colData=metadata,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)
```

Result extraction and visualization

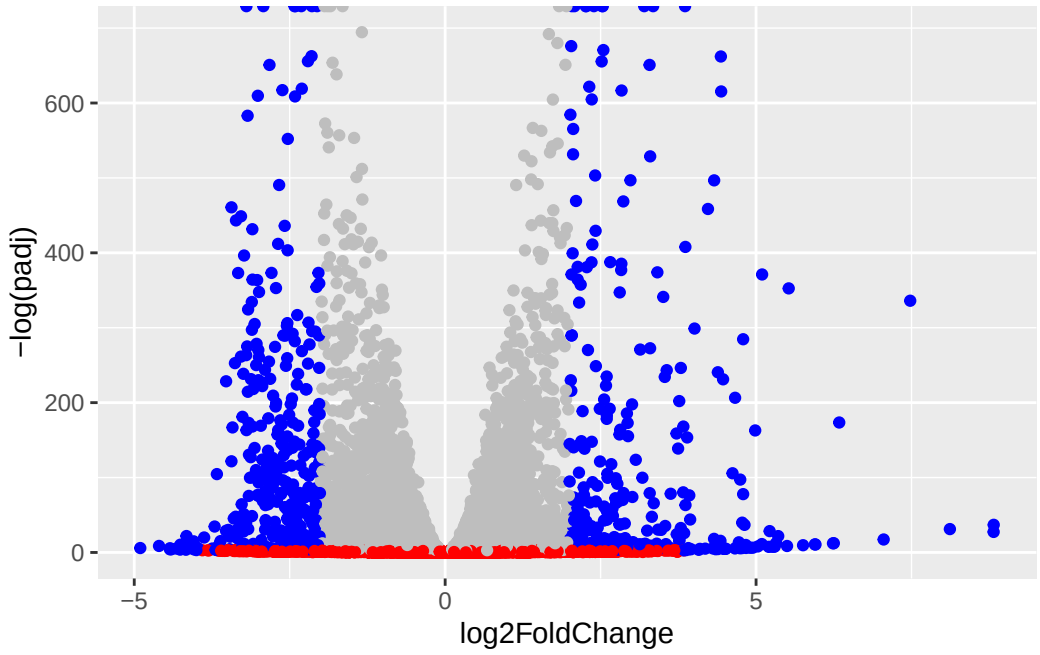
Let's use a volcano plot to show results

```
library(ggplot2)

#Transforming res to a data frame
df <- as.data.frame(res)
mycols<- rep("gray",nrow(df))
mycols[abs(res$log2FoldChange)>2]<- "blue"
mycols[res$padj>=0.05]<- "red"

ggplot(df) + aes(x=log2FoldChange,y=-log(padj)) + geom_point(col=mycols)
```

```
Warning: Removed 1237 rows containing missing values (`geom_point()`).
```



Pathway analysis

```
library("AnnotationDbi")
library("org.Hs.eg.db")
```

Let's see the databases that we can translate between:

```
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"
```

We can now use these "columns" with the mapIds() function to translate between databases identifiers

```
res$symbol<- mapIds(org.Hs.eg.db,keys=row.names(res),keytype="ENSEMBL", column="SYMBOL",mu
```

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

```
res$entrz <- mapIds(org.Hs.eg.db,keys=row.names(res),keytype="ENSEMBL",column="ENTREZID",m
```

```
'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
```

KEGG and GO analysis

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

```
library(gage)
```

```
library(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).
##############################################################################
```

The gage function and wants as input a vector of, in this case, fold changes with names of the genes in a format that matches the database/geneset we are going to use.

```
res = res[order(res$pvalue),]
foldchanges <- res$log2FoldChange
names(foldchanges) <- res$entrz
head(foldchanges)
```

```
     1266     54855      1465     51232      2034      2317
-2.422719  3.201955 -2.313738 -2.059631 -1.888019 -1.649792
```

Kegg pathways

```
data(kegg.sets.hs)
data(sigmet.idx.hs)
# Focus on signaling and metabolic pathways only
kegg.sets.hs = kegg.sets.hs[sigmet.idx.hs]
```

```
# Get the results
keggres = gage(foldchanges, gsets=kegg.sets.hs)
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(foldchanges,pathway.id="hsa04110")
```

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

```
Info: Working in directory C:/Users/Gaby Canto/Desktop/Bioinformatics/Class14
```

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

Section 3. Gene Ontology (GO)

```
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)
```

Have a wee look at the $less here also...

```
head(gobpres$less)
```

```
                                      p.geomean stat.mean        p.val
GO:0048285 organelle fission          1.536227e-15 -8.063910 1.536227e-15
GO:0000280 nuclear division           4.286961e-15 -7.939217 4.286961e-15
GO:0007067 mitosis                    4.286961e-15 -7.939217 4.286961e-15
GO:0000087 M phase of mitotic cell cycle 1.169934e-14 -7.797496 1.169934e-14
GO:0007059 chromosome segregation     2.028624e-11 -6.878340 2.028624e-11
GO:0000236 mitotic prometaphase       1.729553e-10 -6.695966 1.729553e-10
                                             q.val set.size         exp1
GO:0048285 organelle fission          5.843127e-12      376 1.536227e-15
GO:0000280 nuclear division           5.843127e-12      352 4.286961e-15
GO:0007067 mitosis                    5.843127e-12      352 4.286961e-15
GO:0000087 M phase of mitotic cell cycle 1.195965e-11      362 1.169934e-14
GO:0007059 chromosome segregation     1.659009e-08      142 2.028624e-11
GO:0000236 mitotic prometaphase       1.178690e-07       84 1.729553e-10
```

Section 4. Reactome Analysis

We need a list of genes as a text file for using the reactome online site.

Let's start with our genes that have a abs(log2FC) $>$ 2 and a P-value $<$0.05

```
c(T,T) & c(T,F)
```

```
[1]  TRUE FALSE
```

```r
inds <-abs(res$log2FoldChange) >2 & (res$padj < 0.05)
mygenes<-res$symbol[inds]
cat(head(mygenes),sep="\n")
```

```
CNN3
TENT5C
CSRP1
CRIM1
F2RL1
SOX4
```

```r
write.table(mygenes,file="mygenes.txt",quote=FALSE,col.names=FALSE,row.names=FALSE)
```

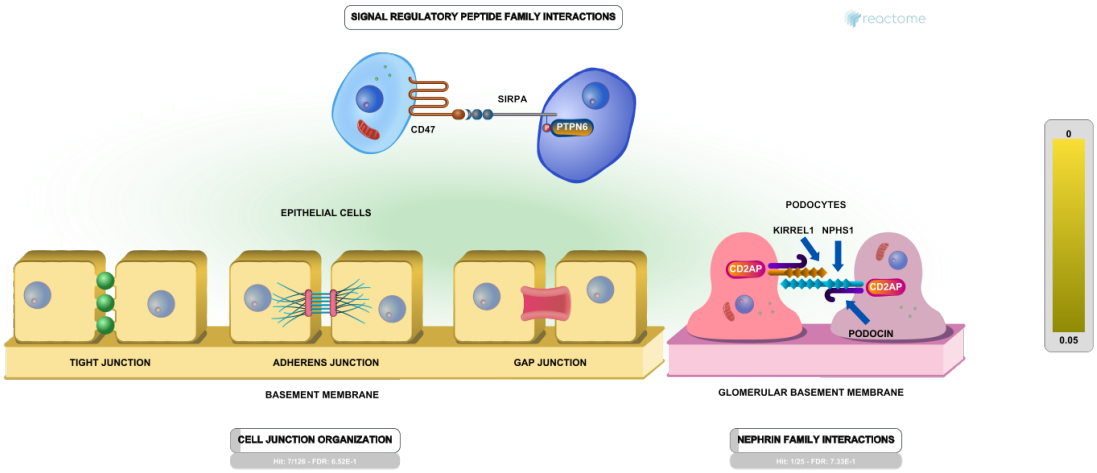


Figure 1: Cell-cell communication