Differential Gene Expression 3 – quantifying differences with Bioconductor
Biol4230 Thurs, April 6, 2018
Bill Pearson wrp@virginia.edu 4-2818 Pinn 6-057

• Bioconductor: a comprehensive 'R' package for expression and genome analysis
  – Obtaining/installing
  – Datasets
  – Vignettes
  – Major packages (affy, edgeR)
• Using Bioconductor/EdgeR for RNAseq
  – reading in data (what to look for)
  – removing genes with low/no signal
  – normalization
  – finding differentially expressed genes

To learn more:

1. Pevsner, Chapter 8 pp. 331-373
bioconductor.org

- More than 600 packages of functions for genome and expression analysis
  - expression analysis: affy
  - RNA-seq: edgeR, DESeq2
  - ChIP-seq (interaction of protein with DNA in chromatin)
  - extracting genomic features
- "Vignettes" that come with data for research problems
- Must be installed (often individually)
- Work with 'R' objects typically much more abstract than data.frames()
- Use the common 'R' logic for selecting rows and columns from data
bioconductor installation

```r
> R ... 'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> source("http://bioconductor.org/biocLite.R")
trying URL 'http://www.bioconductor.org/packages/3.0/bioc/bin/macosx/mavericks/contrib/3.1/BiocInstaller_1.16.2.tgz'
Content type 'application/x-gzip' length 49063 bytes (47 KB)
opened URL

The downloaded binary packages are in
/varfolders/ed/56p6y3_xilq_pr51mdqngcC0000jc/T/\Rtmp8pziwn/downloaded_packages

Bioconductor installs packages incrementally
```

Bioconductor installs packages incrementally

```r
> library(affy)
Error in library(affy) : there is no package called 'affy'
> biocLite('affy')
BioC_Mirror: http://bioconductor.org
Using Bioconductor version 3.0 (BiocInstaller 1.16.2), R version 3.1.3.
Installing package(s) 'affy'
also installing the dependencies 'affyio', 'preprocessCore', 'zlibbioc'
trying URL 'http://bioconductor.org/packages/3.0/bioc/bin/macosx/mavericks/contrib/3.1/affyio_1.34.0.tgz'
Content type 'application/x-gzip' length 89679 bytes (87 KB)
opened URL

The downloaded binary packages are in
/varfolders/ed/56p6y3_xilq_pr51mdqngcC0000jc/T/\Rtmp8pziwn/downloaded_packages
```
Bioconductor installs packages incrementally

```r
> library(affy)
Loading required package: BiocGenerics
Loading required package: parallel
Attaching package: 'BiocGenerics'
The following objects are masked from 'package:parallel':
  ...
  parLapplyLB, parRapply, parSapply, parSapplyLB
The following object is masked from 'package:stats':
  xtabs
The following objects are masked from 'package:base':
  Filter, Find, Map, Position, Reduce, anyDuplicated, append, ...
  unique, unlist, unsplit
Loading required package: Biobase
Welcome to Bioconductor

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

Bioconductor: getting help

![Empirical analysis of digital gene expression data in R](image)

edgeR-package (edgeR)  

R Documentation

Empirical analysis of digital gene expression data in R

Description

edgeR is a package for the analysis of digital gene expression data arising from RNA sequencing technologies such as SAGE, CAGE, Tag-seq or RNA-seq, with emphasis on testing for differential expression.

Particular strengths of the package include the ability to estimate biological variation between replicate libraries, and to conduct exact tests of significance which are suitable for small counts. The package is able to make use of even minimal numbers of replicates.

An extensive User's Guide is available, and can be opened by typing `edgeR:userGuide()` at the R prompt. Detailed help pages are also provided for each individual function.

[Package edgeR version 3.8.6 Index]
Bioconductor: getting help

Documentation

To view documentation for the version of this package installed in your system, start R and enter:

\> browseVignettes("edgeR")

Details

Name: edgeR (version: 3.8.0, date: 2015-06-09)

Description: edgeR is a Bioconductor package for differential expression analysis of digital gene expression data. (Bioinformatics, 26(1), pp. 139-144.)


Bioconductor: getting help

Documentation

To view documentation for the version of this package installed in your system, start R and enter:

\> browseVignettes("edgeR")

Details

Name: edgeR (version: 3.8.0, date: 2015-06-09)

Description: edgeR is a Bioconductor package for differential expression analysis of digital gene expression data. (Bioinformatics, 26(1), pp. 139-144.)


Bioconductor: getting help

Documentation

To view documentation for the version of this package installed in your system, start R and enter:

\> browseVignettes("edgeR")

Details

Name: edgeR (version: 3.8.0, date: 2015-06-09)

Description: edgeR is a Bioconductor package for differential expression analysis of digital gene expression data. (Bioinformatics, 26(1), pp. 139-144.)


Bioconductor: getting help

Documentation

To view documentation for the version of this package installed in your system, start R and enter:

\> browseVignettes("edgeR")

Details

Name: edgeR (version: 3.8.0, date: 2015-06-09)

Description: edgeR is a Bioconductor package for differential expression analysis of digital gene expression data. (Bioinformatics, 26(1), pp. 139-144.)


Bioconductor: getting help

Documentation

To view documentation for the version of this package installed in your system, start R and enter:

\> browseVignettes("edgeR")

Details

Name: edgeR (version: 3.8.0, date: 2015-06-09)

Description: edgeR is a Bioconductor package for differential expression analysis of digital gene expression data. (Bioinformatics, 26(1), pp. 139-144.)

Bioconductor: the installation loop

- Initial intall:
  ```
  source("http://www.bioconductor.org/biocLite.R")
  biocLite()
  ```
- When you need something:
  ```
  > library/simpleaffy)
  Error in library/simpleaffy) : there is no package
called ‘simpleaffy’
  > biocLite("simpleaffy")
  BioC_mirror: http://bioconductor.org
  Using Bioconductor version 3.0 (BiocInstaller 1.16.2),
  Installing package(s) 'simpleaffy'
  also installing the dependencies ‘Biostrings’, ‘gcrma’
  ...
  > library/simpleaffy)
  Loading required package: genefilter
  Attaching package: ‘genefilter’
```

Differential Gene Expression

- Large quantity of data (>20,000 genes)
  - Affychip data has >20 replicates per gene
  - RNAseq has counts (FPKM: Fragments per Kilobase per Million mapped reads)
  - but a small number of biological replicates
- Ideally, identify modest change (1.5x or larger) for modest levels of transcription
  - 10 or fewer transcripts may account for 90% of reads, so 5,000 – 10,000 transcripts for < 10% of reads
  - If technical replicates vary more than 2x, how do you measure 1.5x change?
- Large numbers of tests: how to correct?
  - Family-wide-error-rate (FWER) Bonferroni correction (used for similarity search E()-values)
  - False-discovery-rate (FDR, qvalue)
Identifying differentially expressed genes

1. convert to FPKM (probably not done properly in my example) (cpm)
2. With RNA-seq data, make sure counts > 1
3. Normalize, adjust medians, quantile normalization
4. Look at bulk properties:
   - PCA analysis should group replicates
   - variance should be relatively linear
5. Calculate pair-wise differential expression with t-tests
6. Use topTags to do FDR correction, identify largest changes
   - go back and compare topTags results to actual counts
7. Log2(FC) vs Log10(abundance)
8. Volcano plots show fold-change, q-value tradeoff

Measuring differences – sources of variation

Technical
- RNA isolation
- cDNA synthesis
- hybridization (AffyChip)
- PCR amplification
- G+C content
- sequencing depth
- location on AffyChip/sequencing "lane"

Biological
- genetic background
- sex
- last meal/sleep/exercise
- dividing/quiescent
- cell type within tissue type
- …
Reducing variance improves detection

- Differences in expression in the presence of noise:
  - are the differences significant? statistical model
    - t-test (normally distributed, array data)
    - negative binomial (variance increases with mean)
  - are the differences biological?
    - batch effects from experiment
    - gene effects (length, G+C)
- Analysis packages (edgeR, deSeq2) visualize batch effects, normalize data, apply statistical model
edgeR vs DESeq

Box 2 Differences between DESeq and edgeR

The two packages described in this protocol, DESeq and edgeR, have similar strategies to perform differential analysis for count data. However, they differ in a few important areas.

First, their look and feel differs. For users of the widely used limma package (for analysis of microarray data), the data structures and steps in edgeR follow analogously.

The packages differ in their default normalization: edgeR uses the trimmed mean of M values, whereas DESeq uses a relative log expression approach by creating a virtual library that every sample is compared against; in practice, the normalization factors are often similar.

Perhaps most crucially, the tools differ in the choices made to estimate the dispersion. edgeR moderates feature-level dispersion estimates toward a trended mean according to the dispersion-mean relationship. In contrast, DESeq takes the maximum of the individual dispersion estimates and the dispersion-mean trend.

In practice, this means DESeq is less powerful, whereas edgeR is more sensitive to outliers. Recent comparison studies have highlighted that no single method dominates another across all settings.


Differential Gene Expression with edgeR

- Starting data: HTS counts (not FPKM)
  
  > GSE_HTSeq <- read.table("GSE_ENCODE_HTSeq.txt", 
  + row.names=1,sep='\t',header=T) 
  
  # row.names=1 uses gene names 
  > summary(GSE_HTSeq)

<table>
<thead>
<tr>
<th></th>
<th>GM12892_Rep1</th>
<th>GM12892_Rep2</th>
<th>GM12892_Rep3</th>
<th>H1.hESC_Rep1</th>
<th>H1.hESC_Rep2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min.</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1st Qu.</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Median</td>
<td>103</td>
<td>56.0</td>
<td>47.0</td>
<td>200</td>
<td>208</td>
</tr>
<tr>
<td>3rd Qu.</td>
<td>1246</td>
<td>630.0</td>
<td>567.5</td>
<td>1159</td>
<td>1164</td>
</tr>
<tr>
<td>Mean</td>
<td>1830</td>
<td>814.2</td>
<td>765.1</td>
<td>1418</td>
<td>1460</td>
</tr>
<tr>
<td>Max.</td>
<td>1045434</td>
<td>482679.0</td>
<td>426204.0</td>
<td>646940</td>
<td>628301</td>
</tr>
</tbody>
</table>

  - Do the replicates look similar?
  - Approx how many genes have <= 1 count?
  - Why is the Max 1000X the 3rd quartile?
  - how much data?

  > dim(GSE_HTSeq)
  
  21711 10

Differential Gene Expression with edgeR

- Wide distribution of abundance

\[
\text{summary(GSE_HTS)} \\
\text{GM12892_Rep1} & \text{GM12892_Rep2} & \text{GM12892_Rep3} & \text{H1.hESC_Rep1} & \text{H1.hESC_Rep2} & \text{H1.hESC_Rep3} & \text{H1.hESC_Rep4} & \text{MCF.7_Rep1} & \text{MCF.7_Rep2} & \text{MCF.7_Rep3} & \text{MCF.7_Rep4} \\
\text{Min.} : & 0 & 0 & 0.0 & 0.0 & 0.0 & 0 & 0 & 0 & 0 & 0 \\
\text{1st Qu.:} & 1 & 1 & 1.0 & 1.0 & 1.0 & 1 & 1 & 1 & 1 & 1 \\
\text{Median :} & 103 & 103 & 56.0 & 47.0 & 200 & 200 & 208 & 207 & 208 & 208 \\
\text{Mean :} & 1830 & 1830 & 814.2 & 475.1 & 200 & 200 & 208 & 207 & 208 & 208 \\
\text{3rd Qu.:} & 1246 & 1246 & 630.0 & 567.5 & 1159 & 1164 & 1164 & 1164 & 1164 & 1164 \\
\text{Max. :} & 1045434 & 1045434 & 482679 & 426204 & 646940 & 628301 & 628301 & 628301 & 628301 & 628301 \\
\text{H1.hESC_Rep3} & \text{H1.hESC_Rep4} & \text{MCF.7_Rep1} & \text{MCF.7_Rep2} & \text{MCF.7_Rep3} & \text{MCF.7_Rep4} \\
\text{Min.} : & 0 & 0 & 0.0 & 0.0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\text{1st Qu.:} & 4 & 4 & 2.0 & 2 & 1 & 1 & 1 & 1 & 1 & 1 \\
\text{Median :} & 206 & 206 & 120.0 & 75 & 173 & 173 & 173 & 173 & 173 & 173 \\
\text{Mean :} & 1385 & 1385 & 867.5 & 512.5 & 2450 & 2450 & 2450 & 2450 & 2450 & 2450 \\
\text{3rd Qu.:} & 1130 & 1130 & 713.0 & 530 & 1796 & 1796 & 1796 & 1796 & 1796 & 1796 \\
\text{Max. :} & 597077 & 597077 & 406388 & 2160 & 816273 & 885833 & 885833 & 885833 & 885833 & 885833 \\
\text{colSums(GSE_HTSgt5)} \\
\text{GM12892_Rep1} & \text{GM12892_Rep2} & \text{GM12892_Rep3} & \text{H1.hESC_Rep1} & \text{H1.hESC_Rep2} & \text{H1.hESC_Rep3} & \text{H1.hESC_Rep4} & \text{MCF.7_Rep1} & \text{MCF.7_Rep2} & \text{MCF.7_Rep3} & \text{MCF.7_Rep4} \\
\text{Min.} : & 6 & 6 & 6 & 6 & 6 & 6 & 6 & 6 & 6 & 6 \\
\text{1st Qu.:} & 347 & 347 & 182 & 152 & 341 & 341 & 341 & 341 & 341 & 341 \\
\text{Median :} & 1083 & 1083 & 537 & 487 & 936 & 936 & 936 & 936 & 936 & 936 \\
\text{3rd Qu.:} & 2613 & 2613 & 1333 & 1133 & 2257 & 2257 & 2257 & 2257 & 2257 & 2257 \\
\text{Max. :} & 1045434 & 1045434 & 482679 & 426204 & 646940 & 628301 & 628301 & 628301 & 628301 & 628301 \\
\text{H1.hESC_Rep3} & \text{H1.hESC_Rep4} & \text{MCF.7_Rep1} & \text{MCF.7_Rep2} & \text{MCF.7_Rep3} & \text{MCF.7_Rep4} \\
\text{Min.} : & 6 & 6 & 6 & 6 & 6 & 6 & 6 & 6 & 6 & 6 \\
\text{1st Qu.:} & 342 & 342 & 213 & 152 & 341 & 341 & 341 & 341 & 341 & 341 \\
\text{Median :} & 922 & 922 & 554 & 1542 & 1412 & 1412 & 1412 & 1412 & 1412 & 1412 \\
\text{Mean :} & 2420 & 2420 & 1528 & 9318 & 4072.6 & 4072.6 & 4072.6 & 4072.6 & 4072.6 & 4072.6 \\
\text{3rd Qu.:} & 2195 & 2195 & 1393 & 3642 & 3441.2 & 3441.2 & 3441.2 & 3441.2 & 3441.2 & 3441.2 \\
\text{Max. :} & 1597077 & 1597077 & 406388 & 639987 & 808982 & 885833 & 885833 & 885833 & 885833 & 885833 \\
\text{colSums(GSE_HTSgt5)} \\
\text{GM12892_Rep1} & \text{GM12892_Rep2} & \text{GM12892_Rep3} & \text{H1.hESC_Rep1} & \text{H1.hESC_Rep2} & \text{H1.hESC_Rep3} & \text{H1.hESC_Rep4} & \text{MCF.7_Rep1} & \text{MCF.7_Rep2} & \text{MCF.7_Rep3} & \text{MCF.7_Rep4} \\
\text{37278141} & 16495913 & 15621942 & 28137254 & 28956711 & 27384301 & 17293143 & 44331343 & 46086020 & 4858601
Differential Gene Expression with edgeR

- Convert to a dge (edgeR) structure:
  ```r
  > GSE_dge <- DGEList(counts=GSE_HTSeq, lib.size=colSums(GSE_HTSeq),
  +   group=c(rep("GM128",3),rep("H1",4),rep("MCF7",3)))
  ```
- Select genes with at least n counts
  ```r
  > GSE_cpms <- cpm(GSE_dge)  # cpm, counts per kb per million (FPKM),
  # needs gene lengths
  > keep2 <- rowSums(GSE_cpms[,1:3])>5 & rowSums(GSE_cpms[,4:7]) > 5 &
  rowSums(GSE_cpms[,8:10])>5
  > length(GSE_cpms[keep2,1])
  [1] 10147
  ```
- Set up groups of factors so replicates can be combined:
  ```r
  > GSE_groups <- c(rep("GM128",3),rep("H1",4),rep("MCF7",3))
  > GSE_groups
  [1] "GM128" "GM128" "GM128" "H1" "H1" "H1" "H1" "MCF7" "MCF7" "MCF7"
  ```
Differential Gene Expression with edgeR

- build new dge (edgeR) structure for genes with counts:

```r
> GSE_d2 <- DGEList(counts=GSE_counts2, lib.size=colSums(GSE_counts2), +               group=GSE_groups)
> summary(GSE_counts2[,c(1,2,4,5,8)])
```

```r
GM12892_Rep1   GM12892_Rep2   H1.hESC_Rep1   H1.hESC_Rep2   MCF.7_Rep1
Min. :0     Min. :0     Min. :18     Min. :31.0     Min. :0
1st Qu.:522 1st Qu.:266 1st Qu.:472 1st Qu.:475.5 1st Qu.:741
Median:1274 Median:635  Median:1097 Median:1103.0 Median:1798
Mean :3678 Mean :1624  Mean :2736 Mean :2816.0 Mean :4259
3rd Qu.:2884 3rd Qu.:1353 3rd Qu.:2496 3rd Qu.:2472.5 3rd Qu.:3998
Max. :1045434 Max. :482679 Max. :646940 Max. :628301 Max. :639987
```

- still see differences in bulk properties, but what is 1st quartile now?
- notice that mean is > 3rd quartile. Why?

```r
> summary(GSE_HTseq)
```

```r
GM12892_Rep1   GM12892_Rep2   H1.hESC_Rep1   H1.hESC_Rep2
Min. :0     Min. :0     Min. :200     Min. :208
1st Qu.:1   1st Qu.:4   1st Qu.:1159 1st Qu.:1164
Median:103  Median:208  Median:1159  Median:1164
Mean :1830  Mean :1418  Mean :1460  Mean :1460
3rd Qu.:1246 3rd Qu.:1159 3rd Qu.:1159 3rd Qu.:1164
Max. :1045434 Max. :646940 Max. :646940 Max. :628301
```

Differential Gene Expression with edgeR

- build new dge (edgeR) structure for genes with counts:

```r
> GSE_d2 <- DGEList(counts=GSE_counts2, lib.size=colSums(GSE_counts2), +               group=GSE_groups)
> summary(GSE_counts2[,c(1,2,4,5,8)])
```

```r
GM12892_Rep1   GM12892_Rep2   H1.hESC_Rep1   H1.hESC_Rep2   MCF.7_Rep1
Min. :0     Min. :0     Min. :18     Min. :31.0     Min. :0
1st Qu.:522 1st Qu.:266 1st Qu.:472 1st Qu.:475.5 1st Qu.:741
Median:1274 Median:635  Median:1097 Median:1103.0 Median:1798
Mean :3678 Mean :1624  Mean :2736 Mean :2816.0 Mean :4259
3rd Qu.:2884 3rd Qu.:1353 3rd Qu.:2496 3rd Qu.:2472.5 3rd Qu.:3998
Max. :1045434 Max. :482679 Max. :646940 Max. :628301 Max. :639987
```

- still see differences in bulk properties, but what is 1st quartile now?
- notice that mean is > 3rd quartile. Why?
- Are the bulk properties similar?
How to compare relative mRNA expression?

Figure 1. Normalization and Interpretation of Expression Data


Differential Gene Expression with edgeR

- do some simple normalization, evaluate data quality:

  ```r
  > GSE_d2 <- calcNormFactors(GSE_d2)
  # plot Principal Components Analysis (PCA) of fold-changes
  > plotMDS(GSE_d2, labels=colnames(GSE_counts2),
  + col=c("darkgreen", "red", "blue"[factor(GSE_groups)]))
  > GSE_d2 <- estimateCommonDisp(GSE_d2)
  > GSE_d2 <- estimateTagwiseDisp(GSE_d2)
  > plotMeanVar(GSE_d2, show.tagwise.vars=TRUE, NBline=TRUE)
  > plotBCV(GSE_d2)  # BCV = Biological Coefficient of Variation
  ```
Diversion: Principal Components Analysis (PCA)

- We are interested in the differences, and similarities, between biological samples (replicate treated and controls), from the perspective of expression levels of 10,000 – 20,000 genes.
  - imagine that in the treated sample (e.g. a BHA-treated liver vs normal), only ONE gene has increased expression: (GSTM1), all the rest are the same.
  - to find this gene, we might plot our 6 samples (3 treated, 3 controls) in n=20,000 dimensional space (one axis for every gene), and look so see which point has moved between treated and controls.
  - but if only ONE gene has changed expression level, then all the other genes will be highly correlated, so we do not need 20,000 dimensions, we only need ONE (or possibly two, the second for random noise)
- Principal Components Analysis (PCA) examines the correlation between the datasets, and reduces the dimensionality to the minimum number of "axes" (Principal Components) to explain the variation in the data.
  - first component has most variance – shows a weighting of a gene set that with internal expression correlation, but different from genes not in the set
  - replicate samples should be similar; different samples should be different
  - check for outliers

en.wikipedia.org/wiki/Principal_component_analysis#/media/File:GaussianScatterPCA.png
"GaussianScatterPCA" by — Ben FrantzDale (talk)
### Diversion: Principal Components Analysis (PCA)

```r
> GSE_c2_prin <- princomp(GSE_counts2, cor=T)
> summary(GSE_c2_prin)

Importance of components:

<table>
<thead>
<tr>
<th></th>
<th>Comp.1</th>
<th>Comp.2</th>
<th>Comp.3</th>
<th>Comp.4</th>
<th>Comp.5</th>
<th>Comp.6</th>
</tr>
</thead>
<tbody>
<tr>
<td>SD</td>
<td>2.9115294</td>
<td>0.78123223</td>
<td>0.66206994</td>
<td>0.55342497</td>
<td>0.301356842</td>
<td>0.200813419</td>
</tr>
<tr>
<td>Proc</td>
<td>0.8477003</td>
<td>0.06103238</td>
<td>0.04383366</td>
<td>0.03062792</td>
<td>0.009081595</td>
<td>0.004032603</td>
</tr>
<tr>
<td>Cum</td>
<td>0.8477003</td>
<td>0.90873271</td>
<td>0.95256637</td>
<td>0.98319429</td>
<td>0.992275889</td>
<td>0.996308492</td>
</tr>
</tbody>
</table>
```

![PCA plot](image)

### Differential Gene Expression with edgeR

- Normalization done: do t-tests on gene groups

```r
> GSE_de2 <- exactTest(GSE_d, pair=c("GM128","H1")) # two-way compare
> GSE_tt2 <- topTags(GSE_de2, n=nrow(GSE_de2))
> head(GSE_tt2$table, n=10)

<table>
<thead>
<tr>
<th>gene</th>
<th>logFC</th>
<th>logCPM</th>
<th>PValue</th>
<th>FDR</th>
</tr>
</thead>
<tbody>
<tr>
<td>LCP1</td>
<td>-10.077901</td>
<td>10.155089</td>
<td>8.290374e-57</td>
<td>8.412243e-53</td>
</tr>
<tr>
<td>RASSF5</td>
<td>-6.159025</td>
<td>5.542985</td>
<td>5.657588e-51</td>
<td>2.870377e-47</td>
</tr>
<tr>
<td>TIFA</td>
<td>-5.809657</td>
<td>5.629944</td>
<td>1.409506e-44</td>
<td>4.767419e-41</td>
</tr>
<tr>
<td>SMARCA2</td>
<td>-5.407642</td>
<td>7.584283</td>
<td>3.756655e-43</td>
<td>9.529771e-40</td>
</tr>
<tr>
<td>DOCK10</td>
<td>-5.948203</td>
<td>5.386774</td>
<td>3.129221e-41</td>
<td>6.350442e-38</td>
</tr>
<tr>
<td>CD58</td>
<td>-5.876585</td>
<td>5.406012</td>
<td>1.194831e-40</td>
<td>2.020658e-37</td>
</tr>
<tr>
<td>SMAP2</td>
<td>-5.036339</td>
<td>6.867746</td>
<td>3.039079e-40</td>
<td>4.405362e-37</td>
</tr>
<tr>
<td>NEAT1</td>
<td>-5.451577</td>
<td>9.251840</td>
<td>1.435513e-38</td>
<td>1.820769e-35</td>
</tr>
<tr>
<td>EPHB4</td>
<td>5.930016</td>
<td>7.329484</td>
<td>5.557970e-38</td>
<td>6.266302e-35</td>
</tr>
<tr>
<td>BCL2</td>
<td>-5.237690</td>
<td>5.635830</td>
<td>1.829338e-37</td>
<td>1.856230e-34</td>
</tr>
</tbody>
</table>
```

- what direction are the fold changes? all the same?
- why are PValues < FDR?
Differential Gene Expression with edgeR

- look at "top tags"

```r
> GSE_counts2[row.names(head(GSE_tt$table, n=10)),c(1:3,4:5)]
   GM12892_Rep1 GM12892_Rep2 GM12892_Rep3 H1.hESC_Rep1 H1.hESC_Rep2
LCP1     153736     50863     58206       107       89
RASSF5    5356      2659      1983        63        66
TIFA      3788      2800      2543        91        63
SNARCA2   21764     8381      8004       433       323
DOCK10    4040      2295      2024        77        39
CD58      3145     2201      2219        56        49
SMAP2     12461     5928      4385       317       288
NEAT1     57519    21160     26498      1056       896
EPHB4     149       64        47      7104      7266
BCL2      4073     3012      2634       121       111
```

- Most significant changes have high counts in GM128, low counts in H1.hESC, or vice-versa (EPHB4).
- Go back and look at the data. Does it make sense?

Differential Gene Expression with edgeR

- look at differential expression 1:

```r
> GSE_deg2 <- GSE_rn[GSE_tt$table$FDR < 0.05] # 3879 genes
> GSE_deg2_001 <- GSE_rn[GSE_tt$table$FDR < 0.001] # 1840 genes
> plotSmear(GSE_d2, de.tags=GSE_deg2, main="FDR < 0.05")
> plotSmear(GSE_d2, de.tags=GSE_deg2_001, main="FDR < 0.001")
```
Differential Gene Expression with edgeR

- look at differential expression 2: volcano plots
  
  ```r
  plot(GSE_tt$table$logFC, -log10(GSE_tt$table$FDR),
  + xlab="logFC",ylab="-log10(FDR)"
  ```

look at differential expression 2: volcano plots

```r
plot(GSE_tt$table[,1], -log10(GSE_tt$table[,4]),
  + xlab="logFC",ylab="-log10(FDR)",pch=20)
points(GSE_tt$table[GSE_deg2,1], -log10(GSE_tt$table[GSE_deg2,4]),
  + pch=20,col='red')
points(GSE_tt$table[GSE_deg2_001,1], -log10(GSE_tt$table[GSE_deg2_001,4]),
  + pch=20,col='green')
```
Comprehensive evaluation of differential gene expression analysis methods for RNA-seq data.


fasta.bioch.virginia.edu/biol4230
Identifying differentially expressed genes

1. convert to FPKM (probably not done properly in my example) (cpm)
2. With RNA-seq data, make sure counts > 1
3. Normalize, adjust medians, quantile normalization
4. Look at bulk properties:
   - PCA analysis should group replicates
   - variance should be relatively linear
5. Calculate pair-wise differential expression with t-tests
6. Use topTags to do FDR correction, identify largest changes
   - go back and compare topTags results to actual counts
7. Log2(FC) vs Log10(abundance)
8. Volcano plots show fold-change, q-value tradeoff

Bioconductor: summary

• Bioconductor: a comprehensive 'R' package for expression and genome analysis
  – Obtaining/installing
  – Datasets
  – Vignettes
  – Major packages (affy, edgeR2)
• Using Bioconductor/EdgeR for RNAseq
  – reading in data (what to look for)
  – removing genes with low/no signal
  – normalization
  – finding differentially expressed genes

Always look at the RAW data that produced the list of genes