Analysis of genome-scale count data in Bioconductor

Mark D. Robinson\textsuperscript{1,2} and Davis J. McCarthy\textsuperscript{1}

\textsuperscript{1}Bioinformatics Division, The Walter and Eliza Hall Institute of Medical Research
\textsuperscript{2}Epigenetics Laboratory, Garvan Institute of Medical Research

BioC 2010
Outline

1. Applications
2. Summarization
3. Statistical models for count data
4. “Normalization”
5. Sharing information over entire dataset
6. Statistical testing
7. Other considerations – error model and more complex designs

(Current) Bioconductor tools: baySeq, DEGseq, DESeq, **edgeR**
Sequencing experiments used for:

Sequence of (mapped) read
- e.g. genome sequencing,
- SNP/mutation mapping,
- genomic rearrangements,
- etc.

Position of mapped read
- e.g. RNA-seq, tag-seq for expression,
- ChIP-seq for TF binding or histone modifications,
- MeDIP-seq for DNA methylation,
- etc.
Applications

• **Differential gene expression**: RNA-seq, “Tag”-seq, etc.

• **Differential enrichment**: histone modifications, other types of “enrichment”-based sequencing e.g. ChIP-seq, MeDIP-seq, etc.

• Analyses of **changes** in other tables of counts: e.g. peptide counts from MS/MS experiments, metagenomics experiments.
Example:

RNA-seq (or similar) for gene expression
Example:

Enrichment of subset of the genome (e.g. ChIP for histone modifications or DNA methylation)
Summarization
Summarization

Figure 2

Coding Sequence  Exons  Introns  Splice Junctions
What does genome-scale count data look like?

- e.g. RNA-seq

<table>
<thead>
<tr>
<th>Tag ID</th>
<th>A1</th>
<th>A2</th>
<th>A3</th>
<th>A4</th>
<th>B1</th>
<th>B2</th>
<th>B3</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENSG00000124208</td>
<td>478</td>
<td>619</td>
<td>628</td>
<td>744</td>
<td>483</td>
<td>716</td>
<td>240</td>
</tr>
<tr>
<td>ENSG00000182463</td>
<td>27</td>
<td>20</td>
<td>27</td>
<td>26</td>
<td>48</td>
<td>55</td>
<td>24</td>
</tr>
<tr>
<td>ENSG00000125835</td>
<td>132</td>
<td>200</td>
<td>200</td>
<td>228</td>
<td>560</td>
<td>408</td>
<td>103</td>
</tr>
<tr>
<td>ENSG00000125834</td>
<td>42</td>
<td>60</td>
<td>72</td>
<td>86</td>
<td>131</td>
<td>99</td>
<td>30</td>
</tr>
<tr>
<td>ENSG00000197818</td>
<td>21</td>
<td>29</td>
<td>35</td>
<td>31</td>
<td>52</td>
<td>44</td>
<td>20</td>
</tr>
<tr>
<td>ENSG00000125831</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>ENSG00000215443</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>0</td>
<td>9</td>
<td>7</td>
<td>4</td>
</tr>
<tr>
<td>ENSG00000222008</td>
<td>30</td>
<td>23</td>
<td>29</td>
<td>19</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>ENSG00000101444</td>
<td>46</td>
<td>63</td>
<td>58</td>
<td>71</td>
<td>54</td>
<td>53</td>
<td>17</td>
</tr>
<tr>
<td>ENSG00000101333</td>
<td>2256</td>
<td>2793</td>
<td>3456</td>
<td>3362</td>
<td>2702</td>
<td>2976</td>
<td>1320</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

... tens of thousands more tags ...
Statistical models for count data
Count data

• Count data (e.g. RNA-seq) is discrete, not continuous
• Statistical methods designed for microarrays are not directly applicable
• Two options:
  - Transform count data and apply standard methodology
  - Analyze using models for count data
Count data

- **BUT** we have learned much from the analysis of microarray data
- Methods that share information over the whole dataset generally:
  - stabilize parameter estimation
  - improve performance of making inferences
Poisson arises naturally from multinomial sampling

• Take sample
• Sequence DNA

Library 1

DNA population

Gene 1 \( \lambda_1 \)
Gene 2 \( \lambda_2 \)
Gene 3 \( \lambda_3 \)
Gene 4 \( \lambda_4 \)
Gene 5 \( \lambda_5 \)
Gene 6 \( \lambda_6 \)
...

DNA population
Reads for a single gene (single library) are binomial distributed

\[ Y_i \sim \text{Binomial}(M, \lambda_i) \]

- \( Y_i \) - observed number of reads for gene \( i \)
- \( M \) - total number of sequences
- \( \lambda_i \) - proportion

Large \( M \), small \( \lambda_i \) \( \rightarrow \) approximated well by Poisson( \( \mu_i = M \cdot \lambda_i \) )
Technical replication

- Take another sample from same pool
- Sequence DNA
Poisson replication induces a vuvuzela-shaped “MA”-plot

And the theory validates that this behaviour should exist: M is essentially a log-relative-risk

Power (to detect changes) is higher at higher counts
Implications for downstream analysis.

\[ M_g = \log_2 \frac{Y_{gk} / N_k}{Y_{gk'} / N_{k'}} \]

\[ A_g = \frac{1}{2} \log_2 \left( \frac{Y_{gk} / N_k \cdot Y_{gk'} / N_{k'}}{Y_{gk} / N_k + Y_{gk'} / N_{k'}} \right) \text{ for } Y_g \neq 0 \]
Statistical models

• For count data, variance increases with mean
• Starting point: Poisson model
• Poisson has simplest mean-variance relationship
Poisson

- Variance is equal to the mean
- One-parameter model: mean for each gene

\[ Y_i \sim \text{Pois}(\mu_i) \]

\[ \mu_i = M \times \lambda_i \]

- \( M = \text{library size} \)
- \( \lambda_i = \text{relative contribution of gene} \)
Poisson describes technical variance

- Marioni et al (2008) show that there is little technical variance in RNA-seq
- Poisson model is (probably) adequate for assessing DE when there are only technical reps
- But this is not the end of the story …
Biological replication

2 or more independent DNA populations from the same experimental condition

Generally, experimenters will want biological replication for generalizable results
Overdispersion: extra-Poisson variation

• If there are **ANY** further sources of variation, there is more variation in data than Poisson model can account for

• Poisson model underestimates variation -> false positives

• Need a model that can account for this extra variation
Overdispersion is present in real data

Mean-variance plot for slime-mould dataset hr00 and hr24 (2 vs 2)

Comparing expression levels from Dictyostelium discoideum at hr00 and hr24 – two biological replicates at each time point.
Sources of variation: technical and biological

- Technical: same pool of RNA sequenced separately (e.g. different lanes)
- Biological: RNA from different biological sources (e.g. individuals) under the same experimental conditions
- Other: extra-Poisson variation also introduced by other processes, e.g. different library preparations, protocols etc.
Natural extension to Poisson: negative binomial model

• Introduce the **dispersion parameter**

\[ Y_i \sim NB(\mu_i, \varphi_i) \]

• Still have mean expression level

\[ \mu_i = M \times \lambda_i \]

• \( M = \) library size, \( \lambda_i = \) “conc” of gene DNA

• Variance is a quadratic function of mean:

\[ \text{Var}(Y_i) = \mu_i (1 + \mu_i \varphi_i) \]
Coefficient of variation

- Dispersion is squared coefficient of variation
- Measure of similarity/variability btw samples
- E.g. dispersion = 0.2 -> coef of var = 0.45
- **Interpretation:** true expression levels of genes vary by 45% btw replicates
- Separate biological and technical variation
Problem: small sample size

- RNA-seq experiments will typically have small sample sizes (e.g. n=7)
- Standard methods for estimating the dispersion for each gene produce very unreliable estimates
- Lesson from microarrays: share information between genes (variance structure) to improve inference
Common dispersion model

- One approach: use **same value** for the dispersion for **all genes**
- Estimate using all genes in dataset (conditional max likelihood)
- Produces a reliable estimate
- Nice biological interpretation, but can be heavy handed
Normalization
One particularly powerful advantage of RNA-Seq is that it can capture transcriptome dynamics across different tissues or conditions without sophisticated normalization of data sets\(^{19,20,22}\). RNA-Seq has been

(RPKM) (Fig. 1a,c). The RPKM measure of read density reflects the molar concentration of a transcript in the starting sample by normalizing for RNA length and for the total read number in the measurement. This facilitates transparent comparison of transcript levels both within and between samples.

But, this is not the full story.
Kidney and Liver RNA have very different composition

Robinson and Oshlack (2010) Genome Biology
“Composition” of sampled DNA can be an important consideration

- Hypothetical example: Sequence 6 libraries to the same depth, with varying levels of unique-to-sample counts
- Composition can induce (sometimes significant) differences in counts

Red=low, goldenyellow=high
The adjustment to data analysis is straightforward

- Assumption: core set of genes that do not change in expression.
- Pick a reference sample, compute trimmed mean of M-values (TMM) to reference

\[
\text{LTM}( \frac{Y_{gk}/M_k}{Y_{gk}'/M_k'} ) \text{ estimates } S_k'/S_k
\]

- Adjustment to statistical analysis:
  - Use “effective” library size (edgeR)
  - Use additional offset (GLM)
Outline

1. Applications
2. Summarization
3. Statistical models for count data
4. “Normalization”
5. Sharing information over entire dataset
6. Statistical testing
7. Other considerations – error model and more complex designs

(Current) Bioconductor tools:
baySeq, DEGseq, DESeq, edgeR

Preliminaries (~40min)

Practical (~20min)

More advanced topics (~30min)

Practical (~30min)
Sharing information over entire dataset
Extending the common dispersion model

• Common dispersion offers sig. stabilization vs. naïve tagwise estimation, esp. in small samples.
• Have found common dispersion model to give good results
• **Downside**: not generally true that each tag has the same dispersion.
• Would like stabilized individual tagwise dispersions
Moderated tagwise dispersions

- **Moderate** individual dispersions towards common value
- Stabilize dispersion ests. by sharing variance structure over all genes
- IDEA: ‘Squeeze’ individual dispersion ests. towards common value---larger ests. shrink, smaller ests. get larger
Weighted Likelihood

• WL is the individual log-likelihood plus a weighted version of the common log-likelihood:

$$WL(\phi_g) = l_g(\phi_g) + \alpha l_C(\phi_g)$$

• $l_g$ here is the the quantile-adjusted conditional likelihood

• Plot shows:
  – Black: Likelihood for single tag
  – Blue: Likelihood averaged over all tags (common dispersion)
  – Red: Linear combination of the two

Log-Likelihood

Score (1st derivative of LL)

$$\delta = \frac{\phi}{\phi+1}$$
New alternatives

• DESeq: fit an empirical mean-variance relationship using all data [Anders and Huber 2010]
• baySeq: use all data to form an empirical distribution [Tom Hardcastle]
Statistical testing for count data
Assessing DE: a statistical problem

• Two group setting*: for each gene, estimate $\lambda_1$ and $\lambda_2$ (mean level for each group) and the dispersion

<table>
<thead>
<tr>
<th>Tag ID</th>
<th>A1</th>
<th>A2</th>
<th>A3</th>
<th>A4</th>
<th>B1</th>
<th>B2</th>
<th>B3</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENSG00000215443</td>
<td>14</td>
<td>12</td>
<td>5</td>
<td>13</td>
<td>6</td>
<td>16</td>
<td>14</td>
</tr>
<tr>
<td>ENSG00000222008</td>
<td>97</td>
<td>113</td>
<td>90</td>
<td>101</td>
<td>10</td>
<td>13</td>
<td>10</td>
</tr>
<tr>
<td>ENSG00000101444</td>
<td>46</td>
<td>63</td>
<td>58</td>
<td>71</td>
<td>54</td>
<td>53</td>
<td>1001</td>
</tr>
<tr>
<td>ENSG00000101333</td>
<td>256</td>
<td>793</td>
<td>4156</td>
<td>5463</td>
<td>1705</td>
<td>976</td>
<td>1320</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

• Conduct a hypothesis test for $\lambda_1$ and $\lambda_2$
• Obtain a p-value for the significance of DE for each gene

*Generalises to n groups
Significance testing

• Simple hypothesis test
  \[ H_0: \lambda_1 = \lambda_2 \]
  vs
  \[ H_A: \lambda_1 \neq \lambda_2 \]

• Easy to state, but requires some sophisticated statistics to test appropriately
Multiple testing

• We fit the same model to each gene
• Fit the same model thousands of times
• Expect some (many) genes to appear significantly DE just by chance
• Need to adjust p-values for multiple testing (control the false discovery rate)
• Need accurate p-values to start with
Further considerations

- RNA-seq experiments: very small sample-sizes but need accurate p-values
- Asymptotic tests (Score, Likelihood Ratio, Wald) not ideal
- Instead: exact tests for the Poisson and NB models
- Exact tests give accurate p-values in small sample experiments
Exact testing

• By conditioning on the total sum of counts for each gene we obtain conditional distributions
• Can compute exact p-values from conditional distributions
• Poisson model: sum of Poisson RVs is a Poisson RV
• Conditional distribution (on total sum for a gene) is multinomial
• Two groups: can compute exact p-value for DE from binomial distribution
Exact test for NB distribution

- Sum of NB RVs is a NB RV, if library sizes (means) are equal, under the null hypothesis of no difference
- Conditioning gives ‘overdispersed multinomial’ from which we can compute exact p-values as per binomial test
- Statistical sophistication: quantile-adjustment to equalise library sizes and enable exact test for NB model
- Size of dispersion has big effect on significance of DE
Effect of dispersion

```r
d.tuch$counts[hicom.lotgw,order(d.tuch$samples$group)]
    N8  N33  N51  T8  T33  T51
FABP4  62  62  387   0   37  2022
MMP1   68  74 11190 1883 1998 24955
TTTY15 241   1    0   46    0     0

de.tuch.com$table[hicom.lotgw,]
    logConc  logFC  p.value
FABP4  -15.59  2.016 0.005006
MMP1   -11.59  1.865 0.008713
TTTY15  -17.90 -2.281 0.002998

de.tuch.tgw$table[hicom.lotgw,]
    logConc  logFC  p.value
FABP4  -15.60  2.018 0.05040
MMP1   -11.59  1.866 0.05771
TTTY15  -17.87 -2.238 0.07857

d.tuch$common.dispersion
[1] 0.3325

d.tuch$tagwise.dispersion[hicom.lotgw]
[1] 0.6694 0.6207 0.9417
```
Limitations of exact tests

- Exact tests only implemented for pairwise comparisons between groups
- Can only be used for single-factor (one-dimensional) experimental design
- Cannot include any other factors or covariates in our model for DE
- qCML approach to estimating dispersion also only for single-factor design
Limitations of exact testing

- E.g. cannot account for paired samples in Tuch et al (2010) data
- Matched tumour/normal oral tissue from 3 patients (6 RNA samples)

<table>
<thead>
<tr>
<th>Normal</th>
<th>Tumour</th>
</tr>
</thead>
<tbody>
<tr>
<td>Patient 8</td>
<td>N8</td>
</tr>
<tr>
<td>Patient 33</td>
<td>N33</td>
</tr>
<tr>
<td>Patient 51</td>
<td>N51</td>
</tr>
</tbody>
</table>

Further considerations
More complicated experiments

- We would like to be able to analyse more complicated experimental designs
- Paired samples, time-series, covariates, batch/day effects etc.
- Need to go beyond the qCML and exact tests (sadly)
GLM methods for complicated designs

• Propose to use GLM (generalized linear model) methods for more complicated designs
• Currently implementing likelihood ratio tests
• Cox-Reid approximate conditional inference for estimating dispersion
• Cutting edge…hopefully ready to go soon!
**Example: Cancer dataset**

- RNA-seq data from Tuch et al (2010)
- Comparing oral squamous cell carcinoma tissue to matched healthy oral tissue
- 6 samples, paired design

<table>
<thead>
<tr>
<th>Sample</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>N8</td>
<td>healthy oral tissue from patient 8</td>
</tr>
<tr>
<td>T8</td>
<td>oral tumour tissue from patient 8</td>
</tr>
<tr>
<td>N33</td>
<td>healthy oral tissue from patient 33</td>
</tr>
<tr>
<td>T33</td>
<td>oral tumour tissue from patient 33</td>
</tr>
<tr>
<td>N51</td>
<td>healthy oral tissue from patient 51</td>
</tr>
<tr>
<td>T51</td>
<td>oral tumour tissue from patient 51</td>
</tr>
</tbody>
</table>

*Ignore paired design for now and treat as simple comparison of healthy and tumour groups*
Exact test in edgeR: tagwise disp

> de.tuch.tgw <- exactTest(d.tuch, common.disp=FALSE)
Comparison of groups: tumour - normal
> topTags(de.tuch.tgw, n=5)
Comparison of groups: tumour - normal

<table>
<thead>
<tr>
<th></th>
<th>logConc</th>
<th>logFC</th>
<th>PValue</th>
<th>FDR</th>
</tr>
</thead>
<tbody>
<tr>
<td>TNNC2</td>
<td>-16.63025</td>
<td>-6.439491</td>
<td>6.237545e-12</td>
<td>1.146710e-07</td>
</tr>
<tr>
<td>KRT36</td>
<td>-19.02052</td>
<td>-8.087423</td>
<td>1.723154e-11</td>
<td>1.583923e-07</td>
</tr>
<tr>
<td>ADIPOQ</td>
<td>-19.88465</td>
<td>-7.30664</td>
<td>1.133512e-10</td>
<td>6.946160e-07</td>
</tr>
<tr>
<td>SPP1</td>
<td>-14.90146</td>
<td>6.057058</td>
<td>3.448317e-10</td>
<td>1.288116e-06</td>
</tr>
<tr>
<td>CA3</td>
<td>-15.43170</td>
<td>-6.462589</td>
<td>3.782377e-10</td>
<td>1.288116e-06</td>
</tr>
</tbody>
</table>

> top.tgw <- rownames(topTags(de.tuch.tgw, n=5)$table)
> d.tuch$counts[top.tgw,c(1,3,5,2,4,6)]

<table>
<thead>
<tr>
<th></th>
<th>N8</th>
<th>N33</th>
<th>N51</th>
<th>T8</th>
<th>T33</th>
<th>T51</th>
</tr>
</thead>
<tbody>
<tr>
<td>TNNC2</td>
<td>590</td>
<td>1627</td>
<td>1239</td>
<td>1</td>
<td>8</td>
<td>39</td>
</tr>
<tr>
<td>KRT36</td>
<td>711</td>
<td>104</td>
<td>70</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>ADIPOQ</td>
<td>111</td>
<td>12</td>
<td>575</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>SPP1</td>
<td>19</td>
<td>29</td>
<td>158</td>
<td>378</td>
<td>8517</td>
<td>1681</td>
</tr>
<tr>
<td>CA3</td>
<td>1859</td>
<td>4259</td>
<td>557</td>
<td>1</td>
<td>35</td>
<td>73</td>
</tr>
</tbody>
</table>
## GLM results

```
> glm.res.com[ol[1:10],]

<table>
<thead>
<tr>
<th>Gene</th>
<th>LRT</th>
<th>p-val</th>
<th>N8</th>
<th>N33</th>
<th>N51</th>
<th>T8</th>
<th>T33</th>
<th>T51</th>
</tr>
</thead>
<tbody>
<tr>
<td>TMPRSS11B</td>
<td>9.508e-15</td>
<td>2.609e-13</td>
<td>2601</td>
<td>7874</td>
<td>3399</td>
<td>3</td>
<td>322</td>
<td>9</td>
</tr>
<tr>
<td>TNNC2</td>
<td>2.388e-13</td>
<td>2.609e-13</td>
<td>590</td>
<td>1627</td>
<td>1239</td>
<td>1</td>
<td>8</td>
<td>39</td>
</tr>
<tr>
<td>CKM</td>
<td>2.609e-13</td>
<td>2.609e-13</td>
<td>4120</td>
<td>5203</td>
<td>24175</td>
<td>5</td>
<td>24</td>
<td>1225</td>
</tr>
<tr>
<td>MAL</td>
<td>4.009e-13</td>
<td>4.009e-13</td>
<td>2742</td>
<td>3977</td>
<td>1772</td>
<td>3</td>
<td>264</td>
<td>8</td>
</tr>
<tr>
<td>CRNN</td>
<td>6.646e-13</td>
<td>6.646e-13</td>
<td>24178</td>
<td>22055</td>
<td>12533</td>
<td>49</td>
<td>2353</td>
<td>26</td>
</tr>
<tr>
<td>PI16</td>
<td>6.781e-13</td>
<td>6.781e-13</td>
<td>231</td>
<td>216</td>
<td>1950</td>
<td>0</td>
<td>2</td>
<td>35</td>
</tr>
<tr>
<td>KRT36</td>
<td>2.229e-12</td>
<td>2.229e-12</td>
<td>711</td>
<td>104</td>
<td>70</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>IL1F6</td>
<td>3.513e-12</td>
<td>3.513e-12</td>
<td>367</td>
<td>1825</td>
<td>809</td>
<td>10</td>
<td>45</td>
<td>1</td>
</tr>
<tr>
<td>MYBPC1</td>
<td>3.641e-12</td>
<td>3.641e-12</td>
<td>4791</td>
<td>4145</td>
<td>15766</td>
<td>10</td>
<td>14</td>
<td>1319</td>
</tr>
<tr>
<td>MUC21</td>
<td>1.376e-11</td>
<td>1.376e-11</td>
<td>4161</td>
<td>3432</td>
<td>1722</td>
<td>7</td>
<td>517</td>
<td>5</td>
</tr>
</tbody>
</table>
```
Dispersion estimation

• Estimating the dispersion appropriately for GLMs
→ Cox-Reid approximate conditional inference
Mean-dispersion relationship

- There is evidence of that the value of the dispersion parameter varies with the expression level of the tag
- Noted by Anders and Huber (2010)
- Generally, dispersion is larger for low abundance tags and decreases as abundance increases
Mean-dispersion rel.: ‘t Hoen

Dispersion vs Abundance - ‘t Hoen Data

Abundance (log10(conc))

Dispersion Estimate

-6 -5 -4 -3 -2

0.0 0.1 0.2 0.3 0.4 0.5

common
binned
Also seems true for more datasets
Consequences

- Looks like dispersion is much larger for lower abundance tags.
- Including this in the model would decrease ability to call low abundance tags DE (but further increase power for high abundance tags; is perhaps more correct).
- DESeq has been designed to deal with this.
- edgeR will soon also include an option for allowing dispersion to vary with abundance.
Concluding remarks

- Must understand and account for biological variability (overdispersion) in RNA-seq data
- Negative binomial model, sharing information between genes
- Exact and multiple testing for accurate p-values
References

- Anders and Huber, 2010, Nature Precedings (http://dx.doi.org/10.1038/npre.2010.4282.1)
R Practical
Analysis in R

• R/Bioconductor: open-source statistical software
• Four packages currently available for DE analysis of count data in R
  • DEGSeq (Poisson), edgeR, baySeq and DESeq (NB)
• For NB, variations in the implementation of information sharing and statistical testing
• We work on edgeR, so this is our favourite
Reading in data

• Read the data into R session using a ‘targets’ file
• The function readDGE() creates a ‘DGEList’ object which stores our data in R

```r
> library(edgeR)
> targets <- read.delim(file='Targets.txt',stringsAsFactors=FALSE)
> d <- readDGE(targets,skip=5,comment.char='#')
```
### DGEList object

> d
An object of class "DGEList"

```r
.samples
files group description lib.size
GSM272105 GSM272105.txt DCLK transgenic (Dclk1) mouse hippocampus 2582749
GSM272106 GSM272106.txt WT wild-type mouse hippocampus 3342705
GSM272318 GSM272318.txt DCLK transgenic (Dclk1) mouse hippocampus 3207895
GSM272319 GSM272319.txt WT wild-type mouse hippocampus 3273243
GSM272320 GSM272320.txt DCLK transgenic (Dclk1) mouse hippocampus 2428553
GSM272321 GSM272321.txt WT wild-type mouse hippocampus 358649
GSM272322 GSM272322.txt DCLK transgenic (Dclk1) mouse hippocampus 714498
GSM272323 GSM272323.txt WT wild-type mouse hippocampus 2833329
$counts
```

```r
GSM272105 GSM272106 GSM272318 GSM272319 GSM272320 GSM272321
TTTTTTCTCTTTTCTT 3 1 2 6 3 0
CAGGGACCATCTGTAGA 5 19 2 16 2 0
GTGCCGCTGCAGCTGAGG 7 4 6 5 7 1
ATACACACTGTAAAGAG 2 0 6 4 6 0
AATTATAGTGCAATTGA 5 3 3 3 2 0
GSM272322 GSM272323
TTTTTTCTCTTTTCTT 1 2
CAGGGACCATCTGTAGA 2 13
GTGCCGCTGCAGCTGAGG 2 3
ATACACACTGTAAAGAG 2 8
AATTATAGTGCAATTGA 0 4
76546 more rows ...
```
Multi-dimensional scaling plot

- Used to assess similarity btw libraries - identify outliers and problematic samples
- Common dispersion used as the ‘distance metric’
- Libraries quite similar here, apart from GSM272322

> plotMDS.dge(d)
Estimating the common dispersion

- We now compute common dispersion
- Estimate of the coefficient of variation is 0.44, quite large
- Genuine biological variation so reasonable that there is large inter-library variation

```r
> d <- estimateCommonDisp(d)
> d$common.dispersion
[1] 0.1964033
> sqrt(d$common.dispersion)
[1] 0.4431741
```
Exact test in edgeR: common disp

> de.common <- exactTest(d)
Comparison of groups: WT - DCLK
> topTags(de.common, n=5)
Comparison of groups: WT - DCLK

<table>
<thead>
<tr>
<th>logConc</th>
<th>logFC</th>
<th>PValue</th>
<th>FDR</th>
</tr>
</thead>
<tbody>
<tr>
<td>AATTTCTTCTCTTCTCTCTCTCTT</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TCTGTACGCAGTCAGGC</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CCGTCTTTCTGCTTTGTTCG</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AAGACTCAGGACTCATC</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CCGTCTTTCTGCTTTGTAA</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AATTTCTTCTCTTCTCTCTCTCTT</td>
<td>-17.25</td>
<td>11.671</td>
<td>2.803e-38</td>
</tr>
<tr>
<td>CCGTCTTTCTGCTTTGTTCG</td>
<td>-10.70</td>
<td>5.290</td>
<td>3.524e-22</td>
</tr>
<tr>
<td>AAGACTCAGGACTCATC</td>
<td>-32.22</td>
<td>35.600</td>
<td>1.516e-20</td>
</tr>
<tr>
<td>CCGTCTTTCTGCTTTGTAA</td>
<td>-14.57</td>
<td>5.176</td>
<td>2.716e-20</td>
</tr>
</tbody>
</table>

> d$counts[top.com,order(d$samples$group)]

<table>
<thead>
<tr>
<th>GSM272105</th>
<th>GSM272318</th>
<th>GSM272320</th>
<th>GSM272322</th>
<th>GSM272106</th>
<th>GSM272319</th>
<th>GSM272321</th>
<th>GSM272323</th>
</tr>
</thead>
<tbody>
<tr>
<td>AATTTCTTCTCTTCTCTCTCTCTCTT</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>44</td>
<td>1</td>
<td>76</td>
</tr>
<tr>
<td>TCTGTACGCAGTCAGGC</td>
<td>160</td>
<td>101</td>
<td>440</td>
<td>33</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>CCGTCTTTCTGCTTTGTTCG</td>
<td>106</td>
<td>268</td>
<td>601</td>
<td>5</td>
<td>1485</td>
<td>420</td>
<td>5156</td>
</tr>
<tr>
<td>AAGACTCAGGACTCATC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>6</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>CCGTCTTTCTGCTTTGTAA</td>
<td>12</td>
<td>21</td>
<td>31</td>
<td>1</td>
<td>87</td>
<td>28</td>
<td>352</td>
</tr>
</tbody>
</table>

> sum(topTags(de.common,n=Inf)$table$FDR < 0.01)
[1] 399
Estimating the tagwise dispersions

• One function call required to estimate moderated tagwise dispersions
• The argument ‘prior.n’ determines amount of moderation or ‘squeezing’ towards common disp
• Larger prior.n → more squeezing

> d <- estimateTagwiseDisp(d, prior.n=10)
Using grid search to estimate tagwise dispersion.
> summary(d$tagwise.dispersion)

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
0.119  0.185   0.193   0.197   0.207   0.809
Exact test in edgeR: tagwise disp

```r
> de.tagwise <- exactTest(d, common.disp=FALSE)
Comparison of groups: WT - DCLK
> topTags(de.tagwise, n=5)
Comparison of groups: WT - DCLK

  logConc  logFC  PValue       FDR
CATAAGTCAGAGAGCG  -32.76 -34.508 1.995e-14 7.636e-10
AATTTCTCTCTCTTCT  -17.26  11.668 1.223e-13 3.122e-09
AAAAGAAATCACAGTTG  -32.97 -34.089 6.105e-12 1.168e-07
ATACTGACATTTCTAT  -16.74   4.213 9.744e-12 1.492e-07

> top.tgw <- rownames(topTags(de.tagwise, n=5)$table)
> d$count[[top.tgw,order(d$samples$group)]]

<table>
<thead>
<tr>
<th></th>
<th>GSM272105</th>
<th>GSM272318</th>
<th>GSM272320</th>
<th>GSM272322</th>
<th>GSM272319</th>
</tr>
</thead>
</table>
| TCTGTACGCAGTCAGGC | 160       | 101       | 440       | 33        | 0         | 1
| CATAAGTCAGAGAGCG  | 67        | 77        | 58        | 7         | 0         | 0
| AATTTCTCTCTCTTCT  | 1         | 0         | 0         | 44        | 1         |
| AAAAGAAATCACAGTTG  | 31        | 90        | 42        | 3         | 0         | 0
| ATACTGACATTTCTAT  | 5         | 5         | 8         | 1         | 113       | 228

> sum(topTags(de.tagwise,n=Inf)$table$FDR < 0.01)
[1] 237
```