



Quick note on journal clubs

- Giving feedback to your fellow students is part of your JC mark!
- Feedback is due the following Tuesday (18.00; same as exercises)
- Can be presented remotely or in person (preferred)

How would you rate the presenters' knowledge of the topic? *

Poor
 Fair
 Good
 Very Good
 Excellent

How would you rate the presenters' organization of the subject matter? *

Poor
 Fair
 Good
 Very Good
 Excellent

STA 426 Journal Club feedback

* Required

Select the presentation: *

Choose

Please state your github username (n.b.: this must match the username listed in the markbot reports) *

Your answer

Please write a concise summary of what the article was about (preferably in 2 sentences or less). *

Your answer

Link pinned to #journal_clubs channel



**University of
Zurich**^{UZH}

Statistical Bioinformatics // Institute of Molecular Life Sciences

Goto journal clubs x2



Notes on Exercise 5 (limma fundamentals) .. general things

- please don't hide code (````{r echo=FALSE} ...`!) (even if it's the code given in the assignment)
- try to exclude package startup messages from report, at least for package startup messages (````{r my_block, message=FALSE} ...`)
- solutions are uploaded to `solutions` repo
- Ex 5 marks will be sent by `markbot`



Notes on Exercise 5 (limma fundamentals) .. general things

```
library("affy")  
  
## Loading required package: BiocGenerics  
  
## Loading required package: parallel  
  
##  
## Attaching package: 'BiocGenerics'  
  
## The following objects are masked from 'package:parallel':  
##  
##   clusterApply, clusterApplyLB, clusterCall, clusterEvalQ,  
##   clusterExport, clusterMap, parApply, parCapply, parLapply,  
##   parLapplyLB, parRapply, parSapply, parSapplyLB  
  
## The following object is masked from 'package:limma':  
##  
##   plotMA  
  
## The following objects are masked from 'package:stats':  
##  
##   IQR, mad, sd, var, xtabs
```



Notes on Exercise 5 (limma fundamentals) .. check your report!

```
### Install packages
```

```
r #install.packages("BiocManager") #BiocManager::install('limma') library("limma") library("ggplot2") library(vioplot)
```

Next, we set some parameters for the simulation.

```
## nGenes <- 10000 # number of "features" nSamples <- 6 # number of samples (split equal in 2 groups) pDiff <- .1 # percent of genes "differential grp <- rep(0:1,each=nSampl
```

```
d0 <- 1 s0 <- 0.8 sd <- s0*sqrt(d0/rchisq(nGenes,df=d0)) # dist'n of s.d. ##
```

Next, we can generate a table of (null) data (i.e., no differential features):

```
r y <- matrix(rnorm(nGenes*nSamples, sd=sd), nr=nGenes, nc=nSamples)
```

And, we can add in "differential expression", randomly chosen to be in the positive or negative direction, to a set of indices chosen:

```
r indD <- 1:floor(pDiff*nGenes) diff <- sample(c(-1,1),max(indD),replace=TRUE)*trueFC y[indD,grp==1] <- y[indD,grp==1] + diff
```

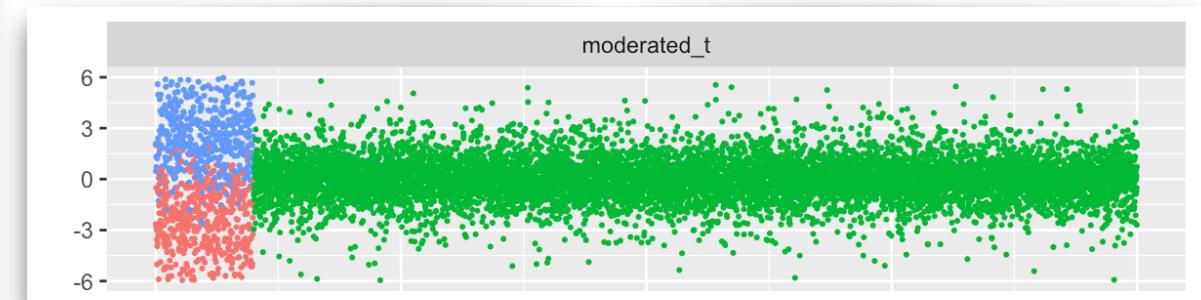
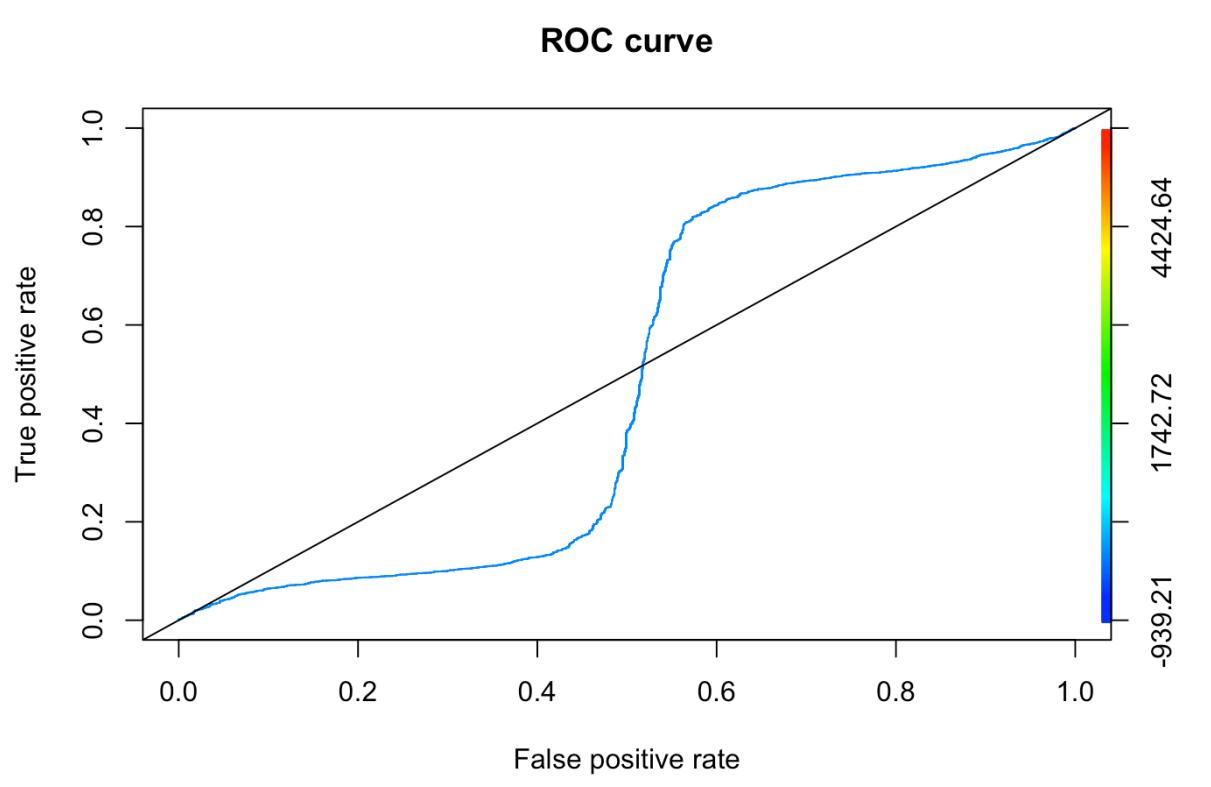


Notes on Exercise 5 (limma fundamentals) .. technical things

- residual variance
- Question asked for: `t.test(..., var.equal=TRUE)`
(Welch's t-test is the default; in the setting here, it's not a paired t-test)
- ROC curve: **absolute value of score** .. or use P-value



Notes on Exercise 5 (limma fundamentals) .. technical things





Expectations: **project**

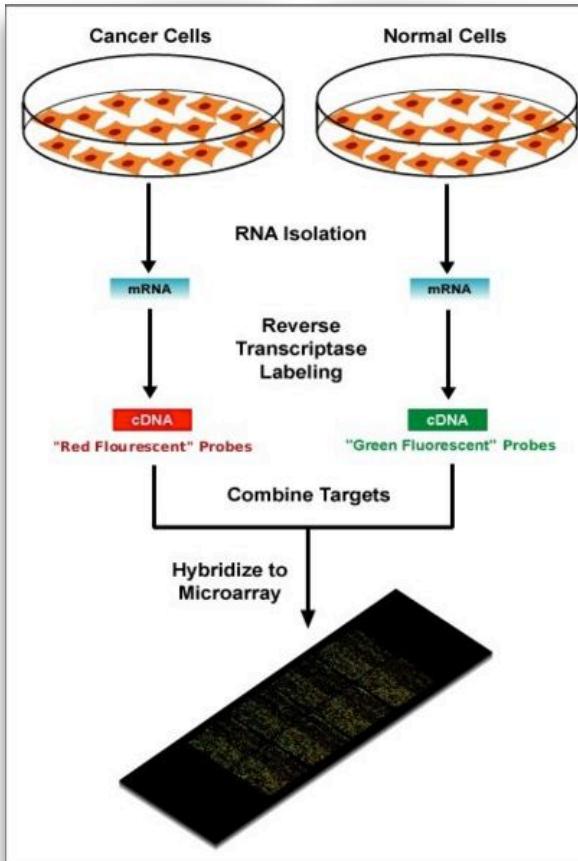
- ~10-15 page report, with R code in line (e.g. **knitr** / **Rmarkdown**)
- Describe the biological setting, statistical analysis, exploratory analysis with publication-quality graphics embedded
- Three possibilities:
 - Comparison of statistical methods (simulation / reference data + metrics)
 - Reproduce an analysis from a paper from the raw data
 - Real collaborative project with FGCZ or a local laboratory
- Be strategic: work on something related to your interests!
- Typically due at end of first working week of January



Statistical models for count data analysis (differential expression)

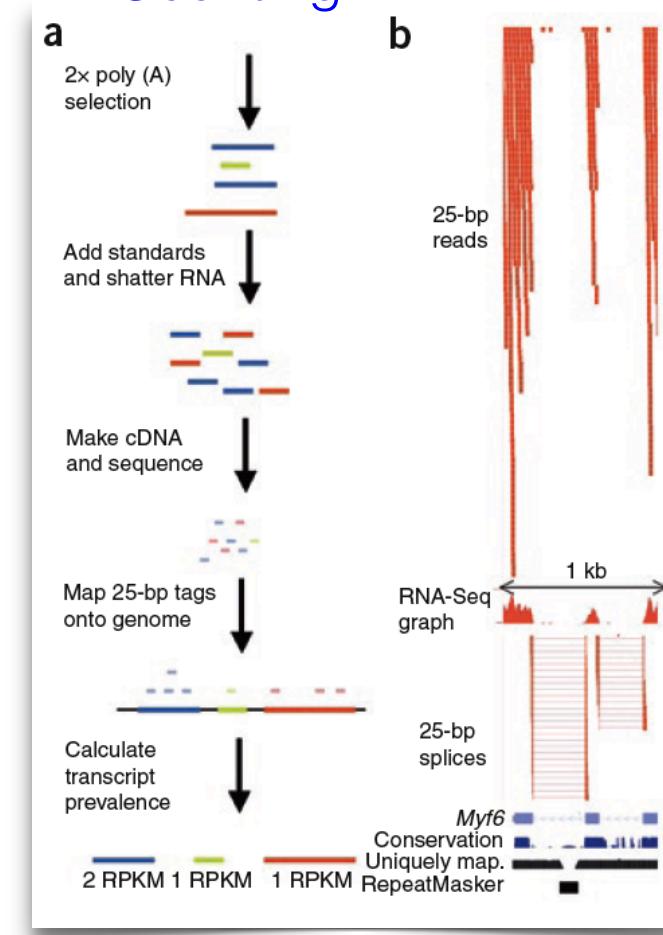
- simple counting (and new alternatives ..)
- edgeR, DESeq/DESeq2 —> why the negative binomial distribution?
- dispersion estimation and information sharing
- normalization considerations
- how about transformations of count data —> limma?

Abundance by Fluorescence Intensity



http://en.wikipedia.org/wiki/DNA_microarray

Abundance by Counting

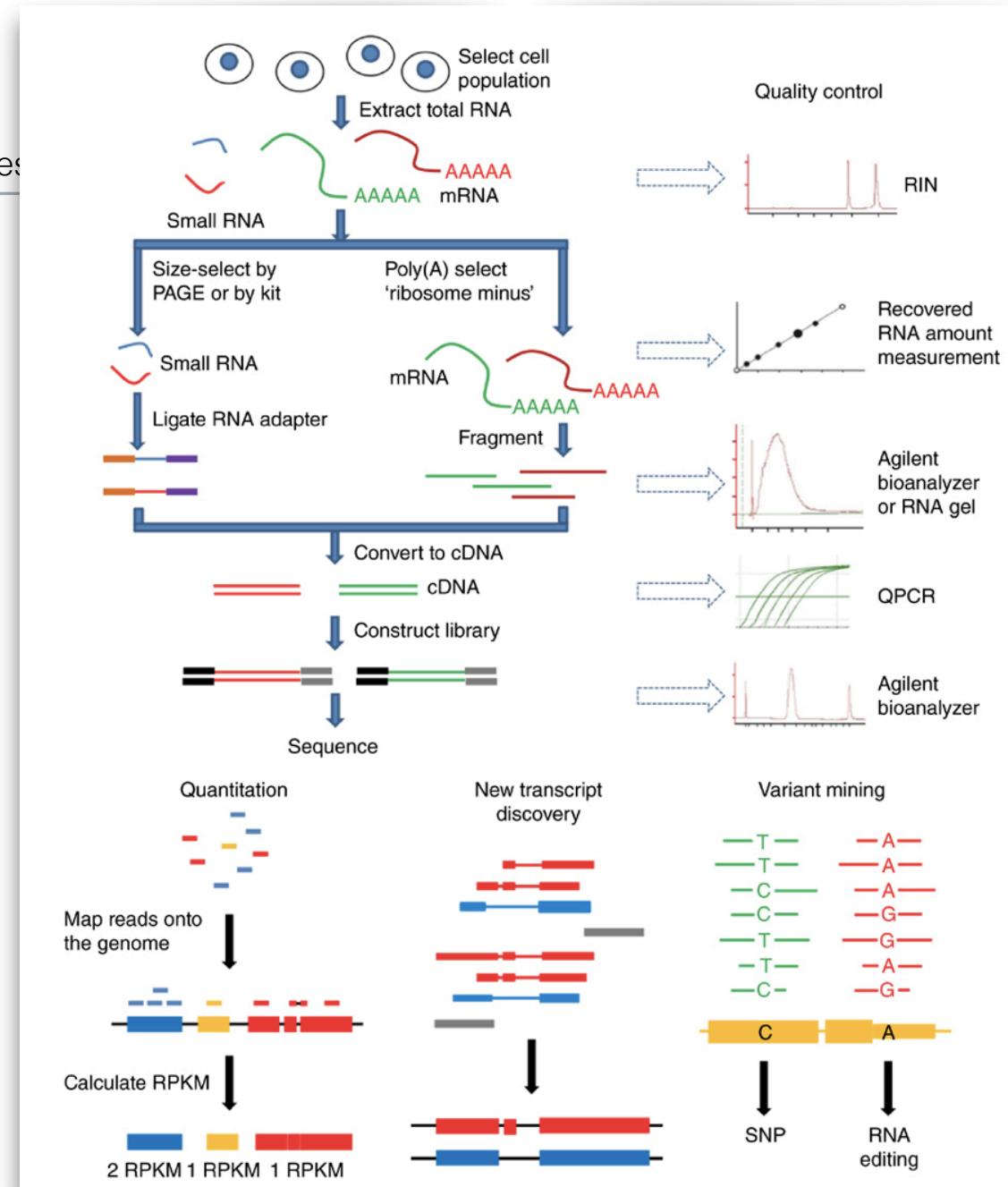


Mortazavi et al., Nature Methods, 2008



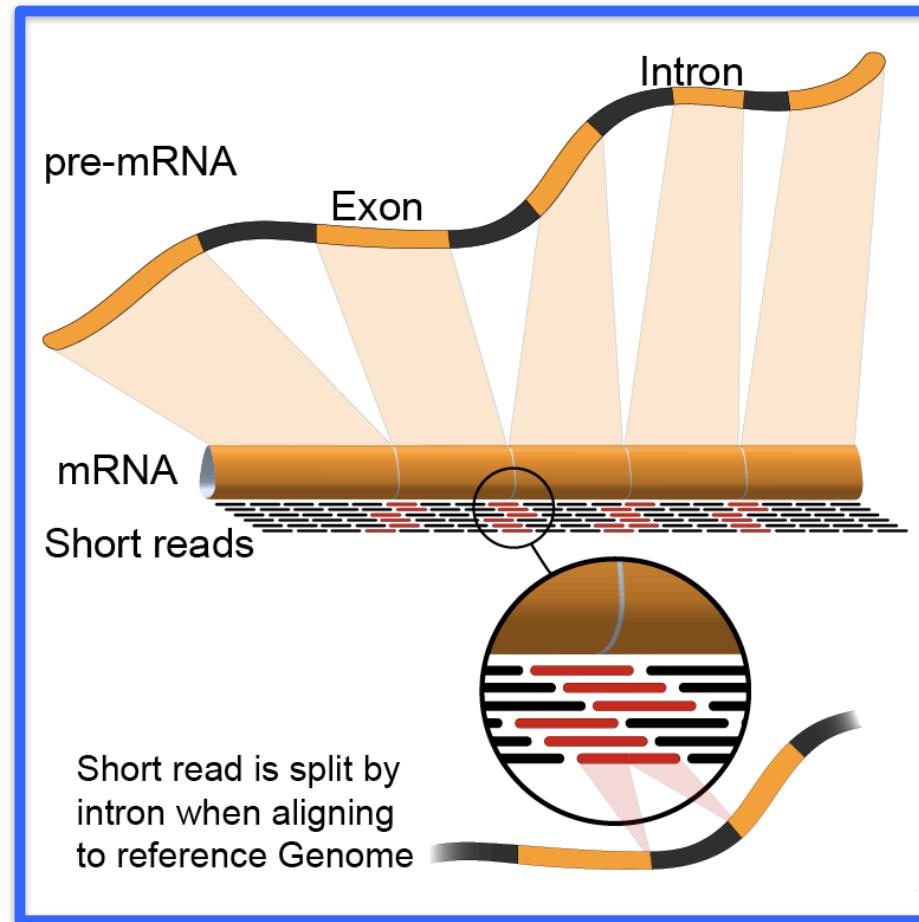
RNA-seq differential expression analyses

1. Map the reads to reference sequences
2. “Count” reads that map to genes (quantify)
3. Compute DE Statistics

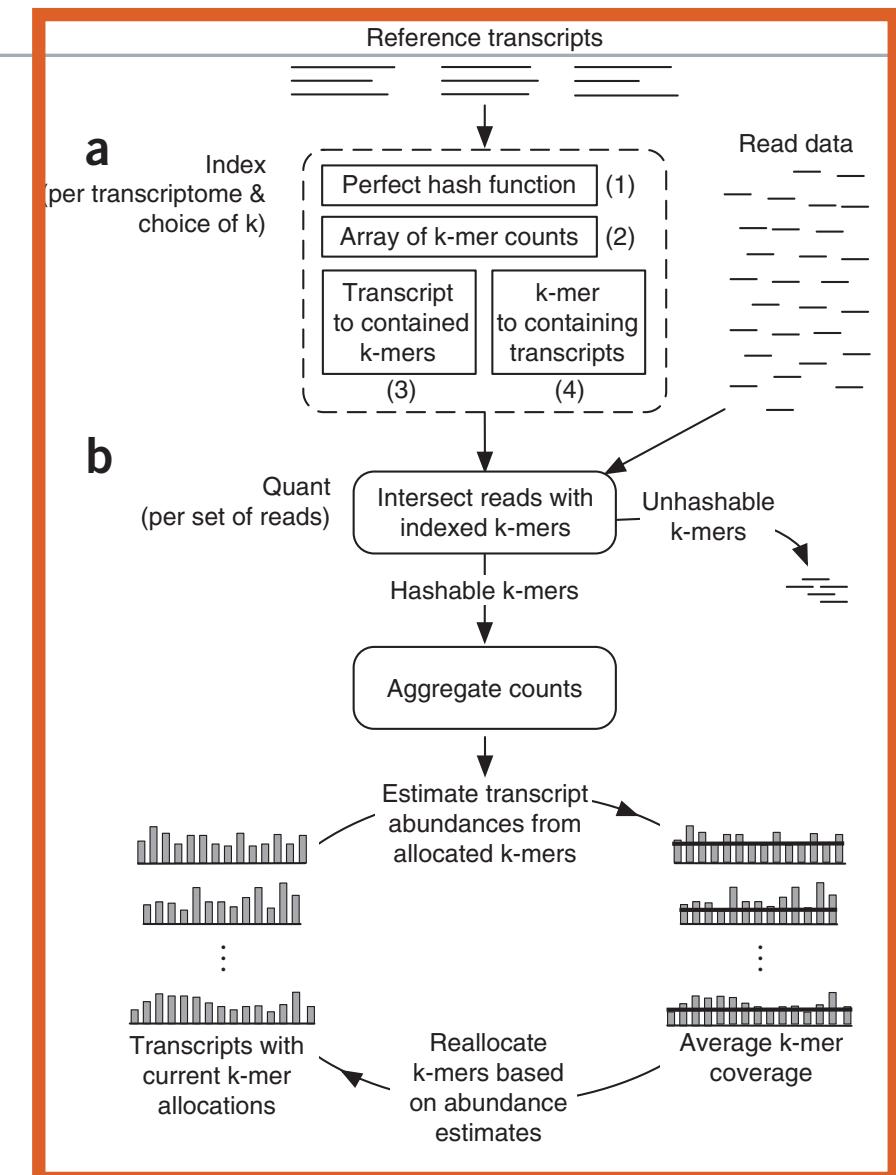


Alignment versus quasi-alignment

Statistical Bioinformatics // Institute of Molecular Life Sciences



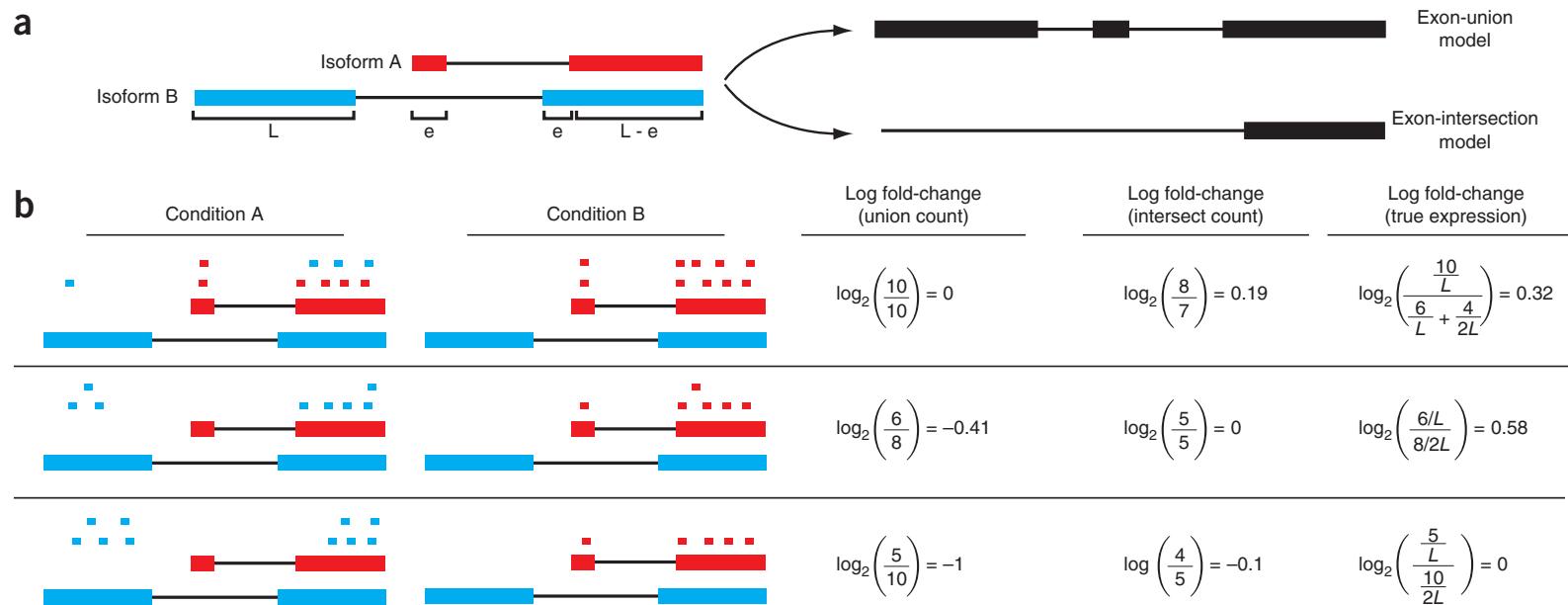
<https://en.wikipedia.org/wiki/RNA-Seq>



sailfish (Patro et al. 2014)

Caveat: simple gene-level counting not perfect, but good first approximation

Trapnell et al. 2013 Nat Biotech



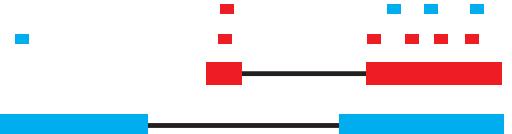
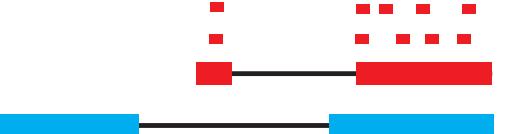
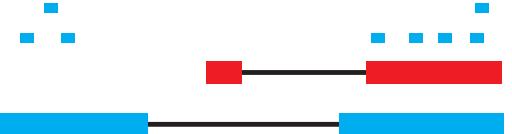
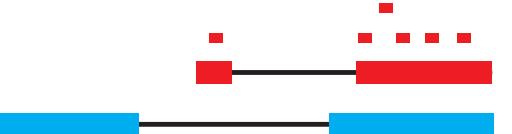
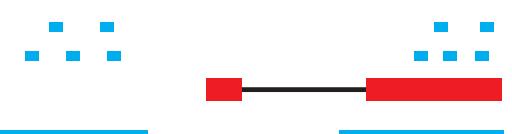
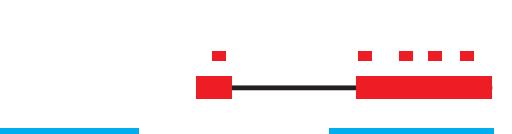
Transcriptome analysis of human tissues and cell lines reveals one dominant transcript per gene

Mar González-Porta¹, Adam Frankish², Johan Rung¹, Jennifer Harrow² and Alvis Brazma^{1*}

Counting/Quantification

union counters → simple sum of all reads
transcript counters → sum of length-normalized reads
(often unknown which reads map to which transcript → portioning)

b

| | Condition A | Condition B | Log fold-change (union count) | Log fold-change (intersect count) | Log fold-change (true expression) |
|--|---|--|--|---|---|
| |  |  | $\log_2\left(\frac{10}{10}\right) = 0$ | $\log_2\left(\frac{8}{7}\right) = 0.19$ | $\log_2\left(\frac{10}{\frac{6}{L} + \frac{4}{2L}}\right) = 0.32$ |
| |  |  | $\log_2\left(\frac{6}{8}\right) = -0.41$ | $\log_2\left(\frac{5}{5}\right) = 0$ | $\log_2\left(\frac{6/L}{8/2L}\right) = 0.58$ |
| |  |  | $\log_2\left(\frac{5}{10}\right) = -1$ | $\log\left(\frac{4}{5}\right) = -0.1$ | $\log_2\left(\frac{5}{\frac{10}{2L}}\right) = 0$ |



How do all these methods of counting affect DE analyses?

You've been doing your RNA-Seq all wrong

Posted by: RNA-Seq Blog in Expression and Quantification November 12, 2015 13,162 Views

In recent years, RNA-seq is emerging as a powerful technology in estimation of gene and/or transcript expression, and RPKM (Reads Per Kilobase per Million reads) is widely used to represent the relative abundance of mRNAs for a gene. In general, the methods for gene quantification can be largely divided into two categories: transcript-based approach and 'union exon'-based approach. Transcript-based approach is intrinsically more difficult because different isoforms of the gene typically have a high proportion of genomic overlap. On the other hand, 'union exon'-based approach method is much simpler and thus widely used in RNA-seq gene quantification. Biologically, a gene is expressed in one or more transcript isoforms. Therefore, transcript-based approach is logically more meaningful than 'union exon'-based approach. Despite the fact that gene quantification is a fundamental task in most RNA-seq studies, however, it remains unclear whether 'union exon'-based approach for RNA-seq gene quantification is a good practice or not.

Researchers at [Pfizer Worldwide Research & Development](#) carried out a side-by-side comparison of 'union exon'-based approach and transcript-based method in RNA-seq gene quantification. It was found that the

F1000Research

F1000Research 2016, 4:1521 Last updated: 05 APR 2016



METHOD ARTICLE

REVISED Differential analyses for RNA-seq: transcript-level estimates improve gene-level inferences [version 2; referees: 2 approved]

Charlotte Soneson^{1,2}, Michael I. Love^{3,4}, Mark D. Robinson^{1,2}

¹Institute for Molecular Life Sciences, University of Zurich, Zurich, 8057, Switzerland

²SIB Swiss Institute of Bioinformatics, University of Zurich, Zurich, 8057, Switzerland

³Department of Biostatistics and Computational Biology, Dana-Farber Cancer Institute, Boston, MA, 02210, USA

⁴Department of Biostatistics, Harvard TH Chan School of Public Health, Boston, MA, 02115, USA

v2 First published: 30 Dec 2015, 4:1521 (doi: [10.12688/f1000research.7563.1](https://doi.org/10.12688/f1000research.7563.1))

Latest published: 29 Feb 2016, 4:1521 (doi: [10.12688/f1000research.7563.2](https://doi.org/10.12688/f1000research.7563.2))

Open Peer Review





Differential expression: why not use methods developed for microarrays?

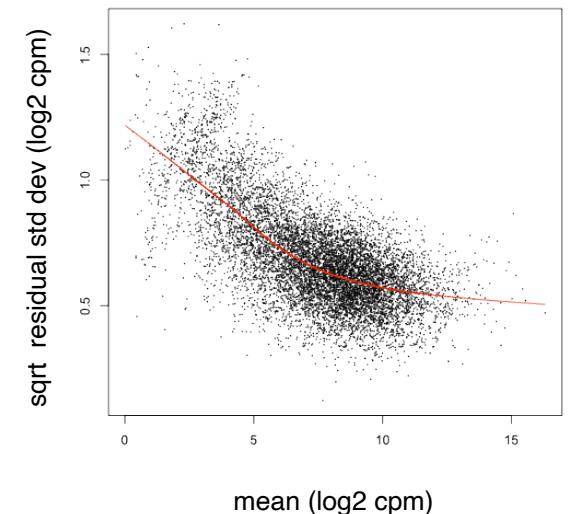
Count data is discrete, not continuous.

Methods designed for microarrays are not directly applicable and suboptimal (**more on this later**)

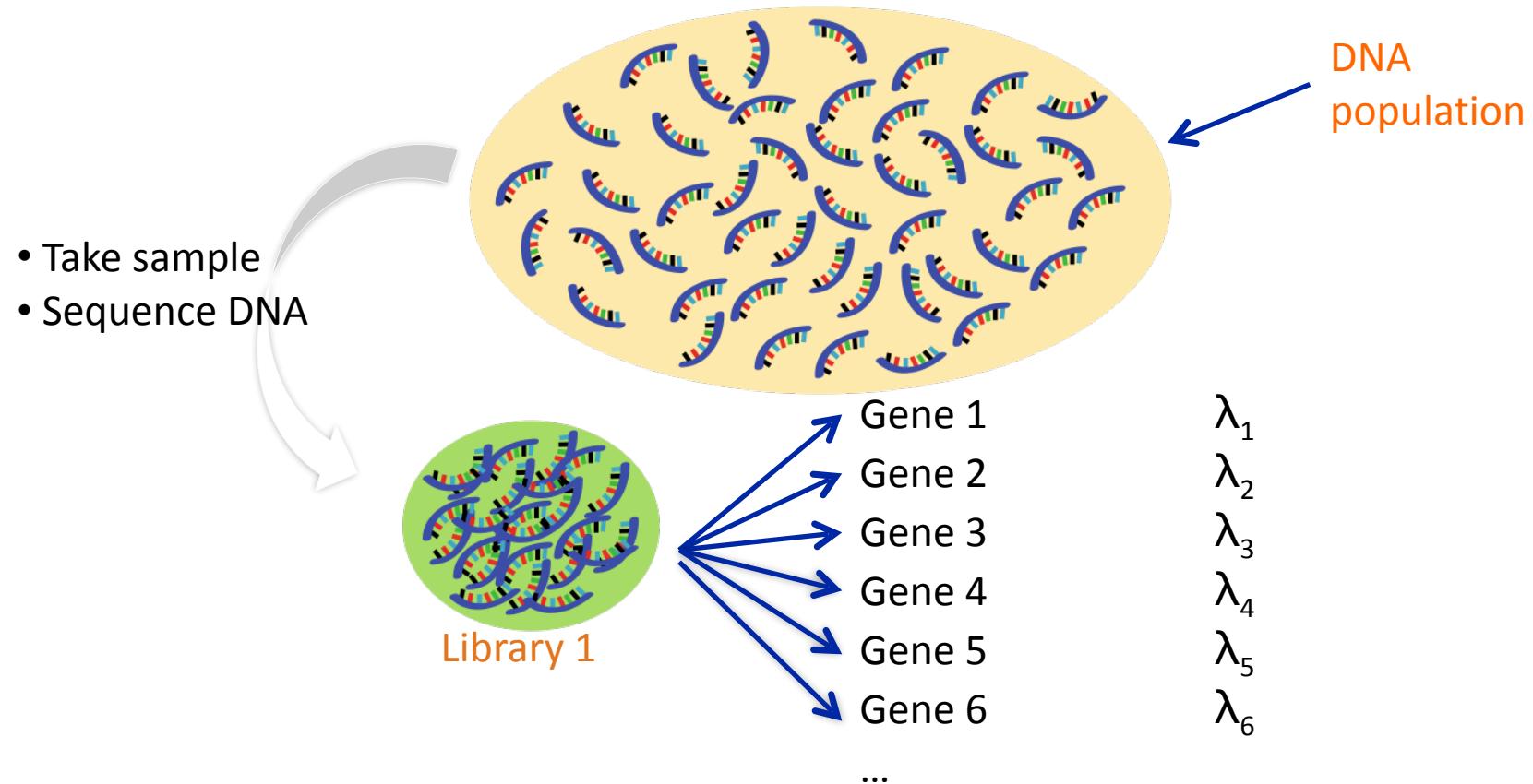
Two options:

Transform count data
and apply standard
methodology

Analyze using
models for count
data



Sampling reads from population of DNA fragments is multinomial



For a single gene, it's a coin toss, i.e. Binomial



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

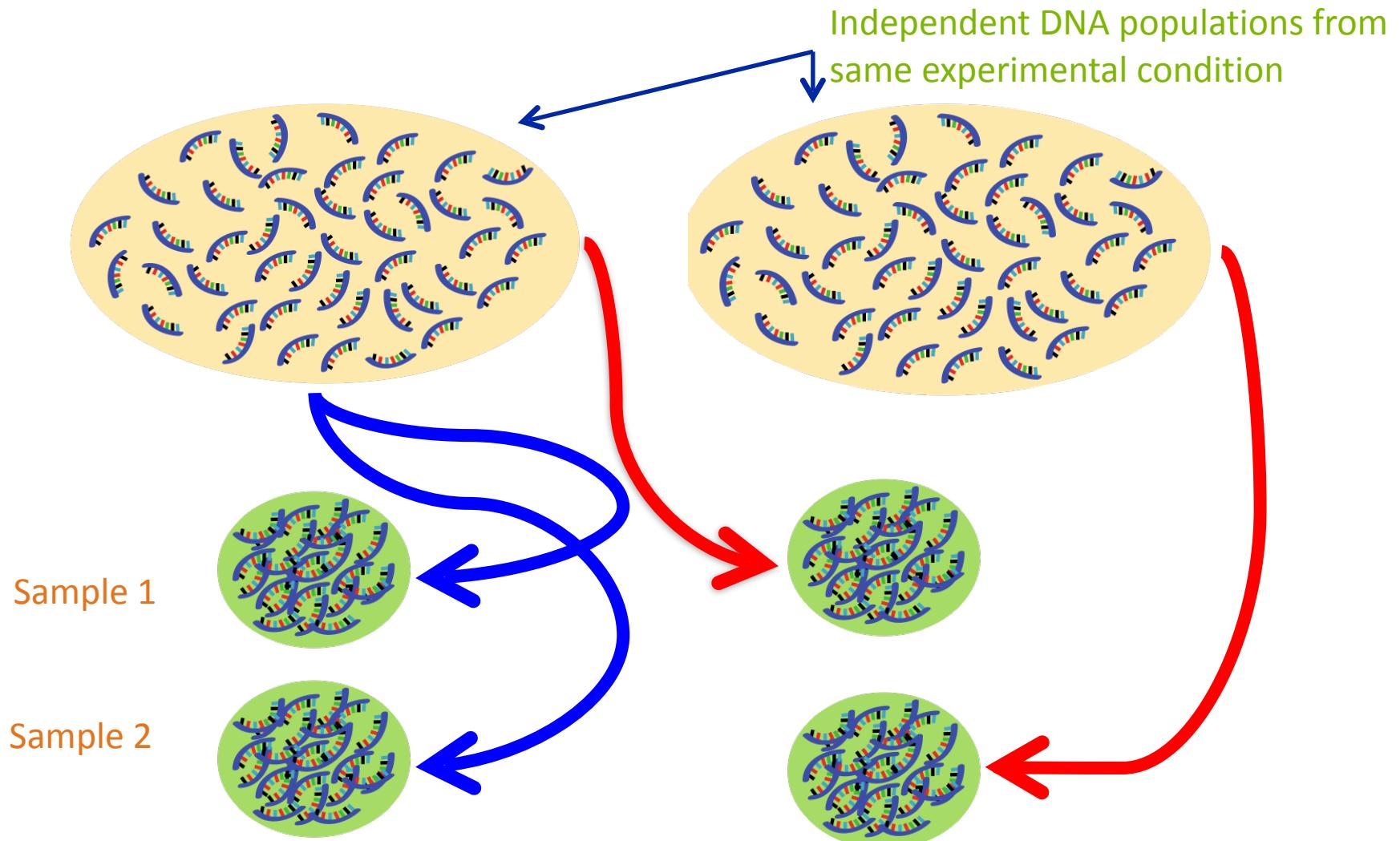
Y_i - observed number of reads for gene i

M - total number of sequences

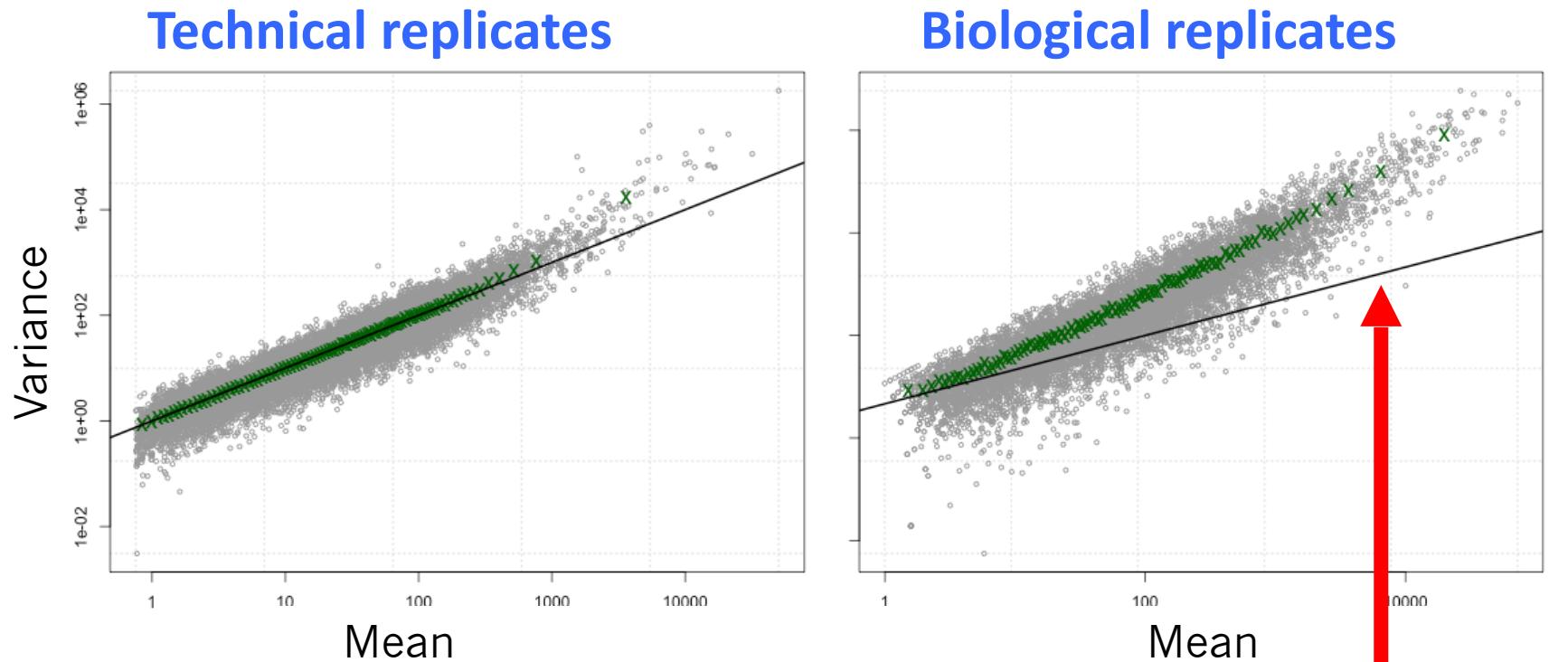
λ_i - proportion

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

Technical replication versus biological replication



Mean-Variance plots: What we see in real data



Data from Marioni et al. *Genome Research* 2008

Data from Parikh et al.
Genome Biology 2010

mean=variance
(Poisson assumption)



Count data modeling assumptions

Poisson adequately describes technical variation

$$Y_i \sim \text{Pois}(M * \lambda_i)$$

$$\text{mean}(Y_i) = \text{variance}(Y_i) = M * \lambda_i$$

Negative binomial (gamma-Poisson) model is a natural extension that allows **biological** variability:

$$Y_i \sim \text{NB}(\mu_i = M * \lambda_i, \phi_i)$$

Same mean, variance is quadratic in the mean:

$$\text{variance}(Y_i) = \mu_i (1 + \mu_i \phi_i)$$

M = library size

λ_i = relative contribution of gene i



Similar interpretation

$$Y_i \sim NB(\mu_i = N_i * \lambda_i, \phi_i)$$

$$E(y_{gi}) = \mu_{gi} = N_i \pi_{gi}.$$

(Coefficient of variation = standard deviation/mean)

$$\text{var}(y_{gi}) = E_\pi[\text{var}(y|\pi)] + \text{var}_\pi[E(y|\pi)] = \mu_{gi} + \phi_g \mu_{gi}^2.$$

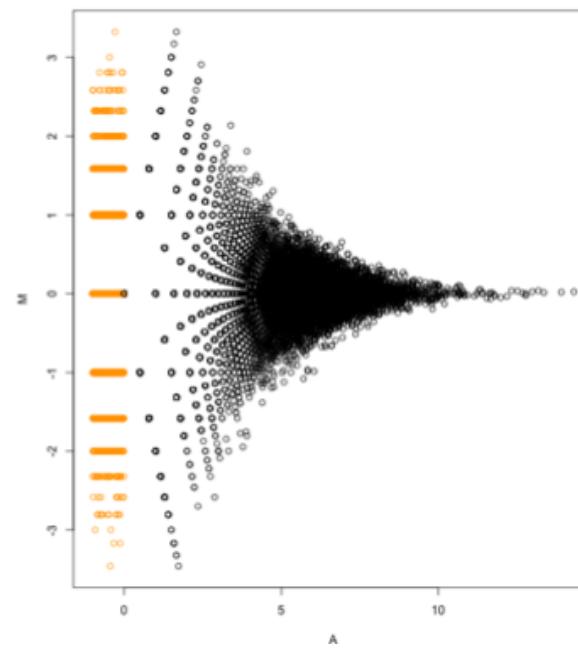
Dividing both sides by μ_{gi}^2 gives

$$CV^2(y_{gi}) = 1/\mu_{gi} + \phi_g.$$

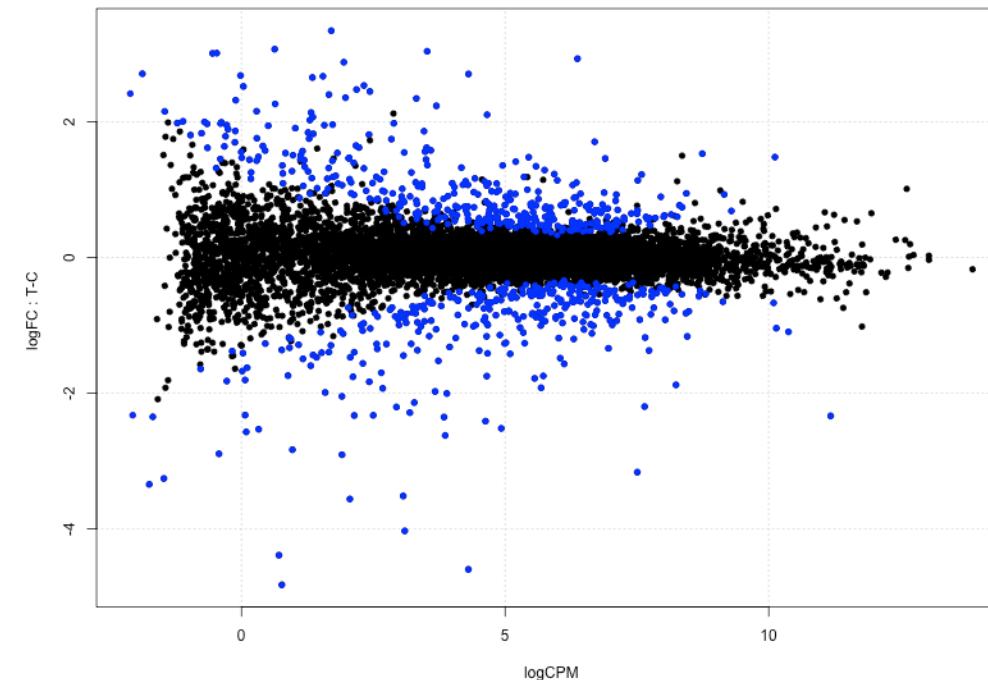
$$CV^2(\gamma_{gi}) = 1/\mu_{gi} + \phi_g.$$

A confirmation of what the theory states

Technical replicates
(~Poisson)



Biological replicates





Differential expression, small sample inference —> **except now with counts**

- Table of data (e.g., microarray gene expression data with replicates of each of condition A, condition B)
 - rows = features (e.g., genes), columns = experimental units (samples)
- Most common problem in statistical bioinformatics: want to infer whether there is a change in the response
—> a statistical test for each row of the table.

What test might you use? Why is this hard? What issues arise? How much statistical power is there [1] ?

| > head(y) | group0 | group0 | group0 | group1 | group1 | group1 |
|-----------|------------|------------|-------------|------------|------------|-------------|
| gene1 | -0.1874854 | 0.2584037 | -0.05550717 | -0.4617966 | -0.3563024 | -0.03271432 |
| gene2 | -3.5418798 | -2.4540999 | 0.11750996 | -4.3270442 | -5.3462622 | -5.54049106 |
| gene3 | -0.1226303 | 0.9354707 | -1.10537767 | -0.1037990 | 0.5221678 | -1.72360854 |
| gene4 | -2.3394536 | -0.3495697 | -3.47742610 | -3.2287093 | 6.1376670 | -2.23871974 |
| gene5 | -3.7978820 | 1.4545702 | -7.14796503 | -4.0500796 | 4.7235714 | 10.00033769 |
| gene6 | 1.4627078 | -0.3096070 | -0.26230124 | -0.7903434 | 0.8398769 | -0.96822312 |



What was successful with microarray data: classical/moderated/shrunken t-tests

$$t_g = \frac{\bar{y}_{\text{mu}} - \bar{y}_{\text{wt}}}{s_g c}$$

Feature-specific

$$\tilde{t}_g = \frac{\bar{y}_{\text{mu}} - \bar{y}_{\text{wt}}}{\tilde{s}_g u}$$

Moderated

$$t_{g,\text{pooled}} = \frac{\bar{y}_{\text{mu}} - \bar{y}_{\text{wt}}}{s_0 c}$$

Common



Let's try the same strategy with counts

At one extreme, assume all genes have same dispersion (too strong)

At other extreme, estimate dispersion separately/independently for each gene (poor estimates)

Shrink individual estimates toward common/trend (how?)

No hierarchical model (e.g. limma) to do this —> **approximations,
weighted likelihood**

No t-distribution theory to formulate statistical tests.



Count data modeling assumptions

Poisson adequately describes technical variation

$$Y_i \sim \text{Pois}(M * \lambda_i)$$

$$\text{mean}(Y_i) = \text{variance}(Y_i) = M * \lambda_i$$

Negative binomial (gamma-Poisson) model is a natural extension that allows **biological** variability:

$$Y_i \sim \text{NB}(\mu_i = M * \lambda_i, \phi_i)$$

Same mean, variance is quadratic in the mean:

$$\text{variance}(Y_i) = \mu_i (1 + \mu_i \phi_i)$$

M = library size

λ_i = relative contribution of gene i



First challenge: getting good estimates of dispersion in small samples

Several choices here:

- Maximum Likelihood (MLE)
- Pseudo-Likelihood (PL)
- Quasi-Likelihood (QL)
- Conditional Maximum Likelihood (CML)
- Approximate Conditional Inference (Cox-Reid)
- *quantile-adjusted Maximum Likelihood (qCML)*

$$Y_{gij} \sim \text{NegBin}(\mu_{gi} = M_j \lambda_{gi}, \phi)$$

$$(\hat{\lambda}_{MLE}, \hat{\phi}_{MLE}) \big| = \arg \max_{\lambda, \phi} l(\lambda, \phi)$$

$$X^2 = \sum_{gij} \frac{(y_{gij} - \hat{\mu}_{gi})^2}{\hat{\mu}_{gi}(1 + \hat{\phi}_{PL}\hat{\mu}_{gi})} = G(n_1 + n_2 - 2)$$

$$D = 2 \sum_{gij} \left\{ y_{gij} \log \left[\frac{y_{gij}}{\mu_{gi}} \right] - (y_{gij} + \phi_{QL}^{-1}) \log \left[\frac{y_{gij} + \phi_{QL}^{-1}}{\mu_{gi} + \phi_{QL}^{-1}} \right] \right\}$$



Conditional likelihood

Likelihood for single **negative binomial** observation:

$$f(y; \mu, \phi) = P(Y = y) = \frac{\Gamma(y + \phi^{-1})}{\Gamma(\phi^{-1})\Gamma(y + 1)} \left(\frac{1}{1 + \mu\phi} \right)^{\phi^{-1}} \left(\frac{\mu}{\phi^{-1} + \mu} \right)^y$$

If all libraries are the same size (i.e. $m_i \equiv m$), the sum $Z = Y_1 + \dots + Y_n \sim \text{NB}(nm\lambda, \phi n^{-1})$

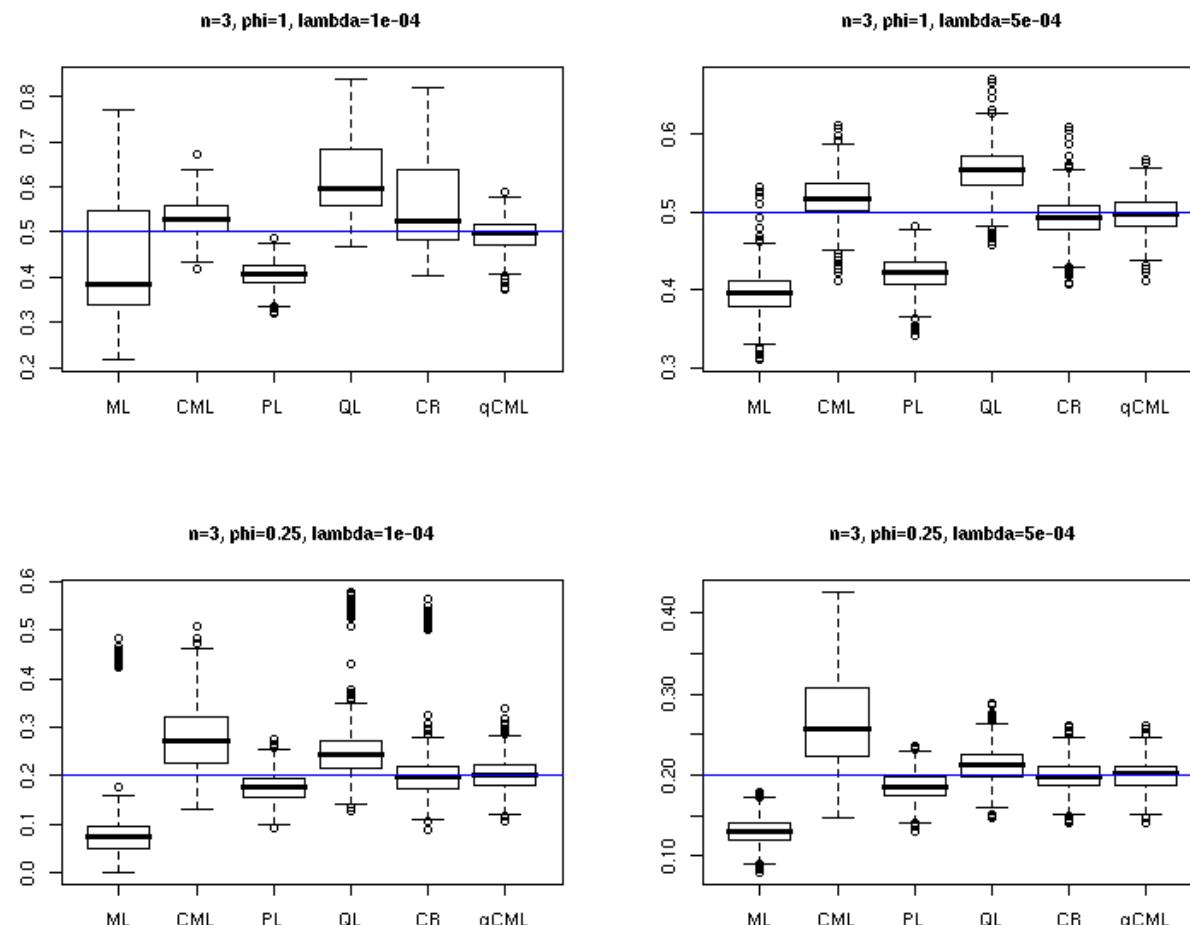
Thus, can form conditional likelihood:

$$l_{Y|Z=z}(\phi) = \left[\sum_{i=1}^n \log \Gamma(y_i + \phi^{-1}) \right] + \log \Gamma(n\phi^{-1}) - \log \Gamma(z + n\phi^{-1}) - n \log \Gamma(\phi^{-1})$$

Comparison of Estimators (Common Dispersion)

Horizontal blue line is
TRUE value.

qCML performs best
under a wide range of
conditions.





Likelihood —> Weighted likelihood

Likelihood:
$$L(X; \theta) = \prod_i^n f(x_i; \theta)$$

log-likelihood:

$$l(X; \theta) = \log(L(X; \theta)) = \sum_i^n \log(f(x_i; \theta))$$

MLE:
$$\hat{\theta} = \arg \max_{\theta} l(X; \theta)$$



Likelihood —> Weighted likelihood

$$WL(X; \theta) = \prod_i^n f(x_i; \theta)^{w_i}, \text{ where } w_i \text{ is weight.}$$

$$wl(X; \theta) = \log(WL(X; \theta)) = \sum_i^n w_i \log(f(x_i; \theta))$$

$$\hat{\theta} = \arg \max_{\theta} wl(X; \theta)$$



Second challenge: Moderate dispersion estimate

Weighted likelihood -- individual log-likelihood plus a weighted version of the common log-likelihood:

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

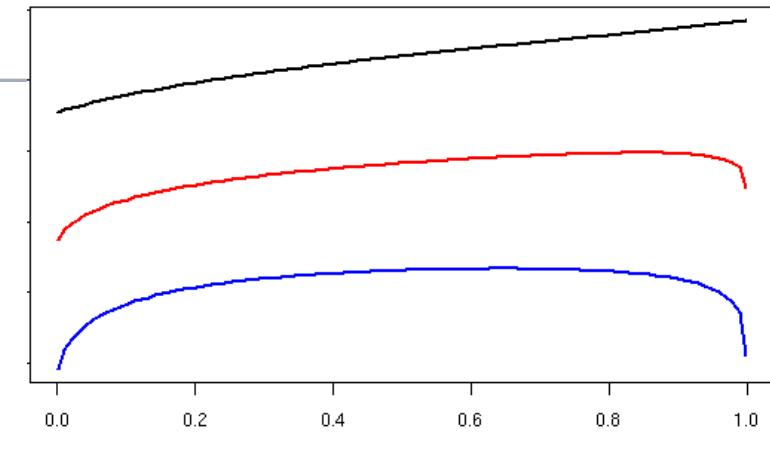
l_g - quantile-adjusted conditional likelihood

Black: single tag

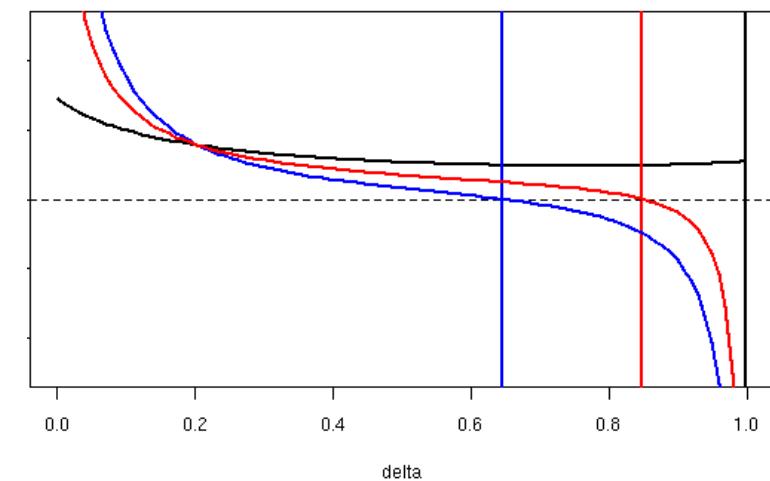
Blue: common dispersion

Red: Linear combination of the two

Log-Likelihood



Score (1st derivative of LL)



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

How much to shrink?

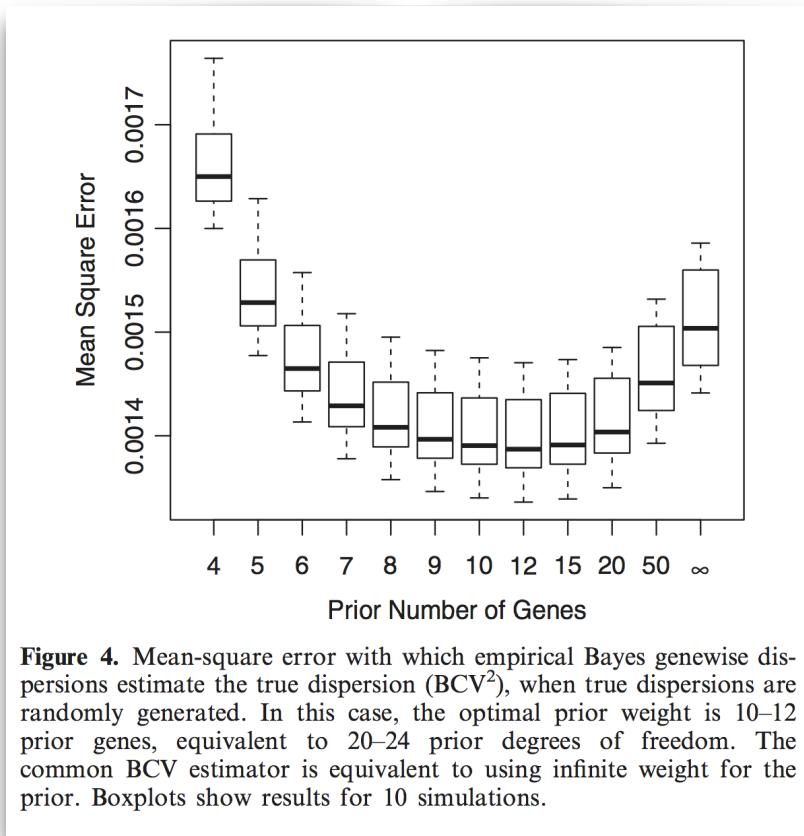


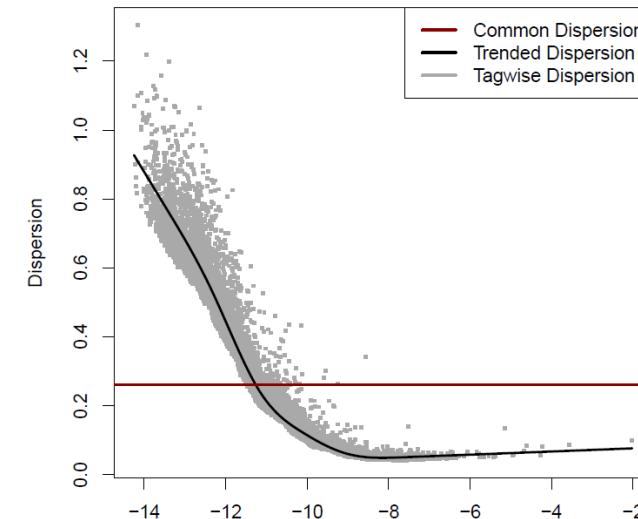
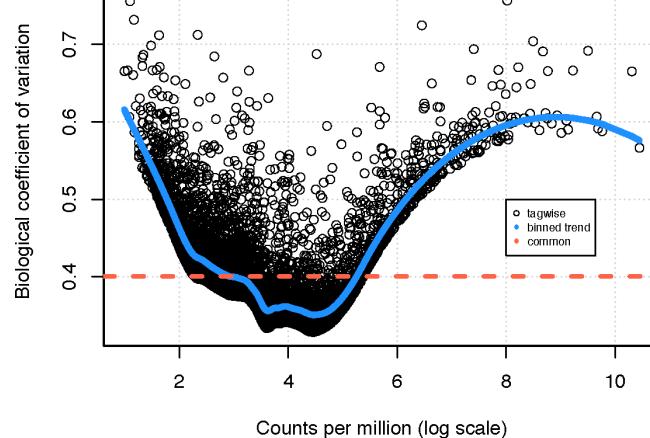
Figure 4. Mean-square error with which empirical Bayes genewise dispersions estimate the true dispersion (BCV^2), when true dispersions are randomly generated. In this case, the optimal prior weight is 10–12 prior genes, equivalent to 20–24 prior degrees of freedom. The common BCV estimator is equivalent to using infinite weight for the prior. Boxplots show results for 10 simulations.

Simulations suggest there is an optimal amount to shrink.

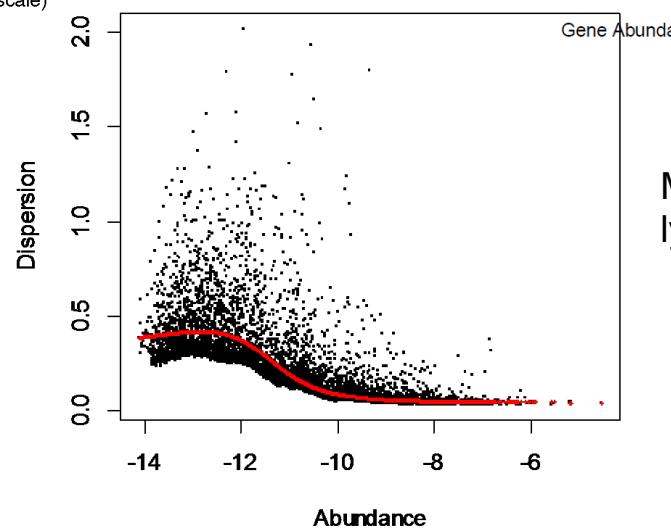
Challenge: choosing/estimating how much

Dispersion varies with mean: moderate dispersion towards **trend**

Data:
Tuch et al.,
2008



Mouse hematopoietic
stem cells



Mouse
lymphomas

Advantage: genes are allowed to have their own variance.



Nature Reviews Genetics | AOP, published online 18 November 2008; doi:10.1038/nrg2484

INNOVATION

RNA-Seq: a revolutionary tool for transcriptomics

Zhong Wang, Mark Gerstein and Michael Snyder

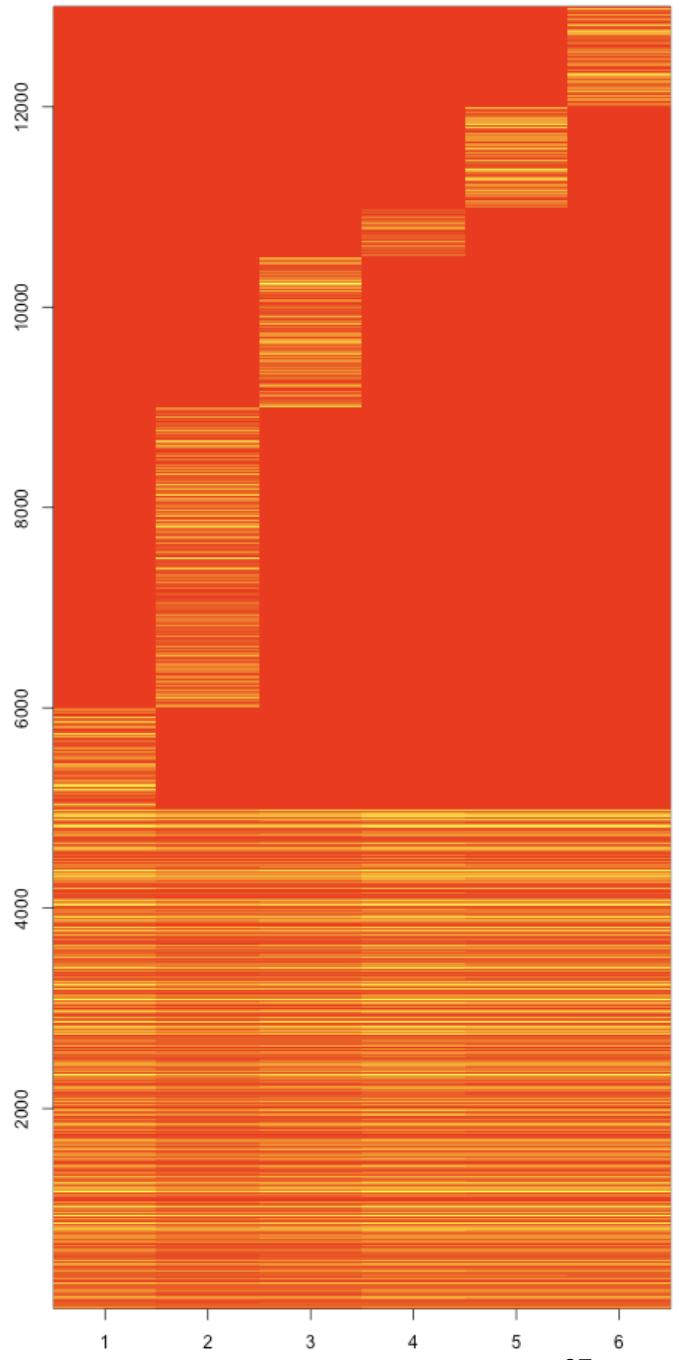
One particularly powerful advantage of RNA-Seq is that it can capture transcriptome dynamics across different tissues or conditions without sophisticated normalization of data sets^{19,20,22}.



“Composition” or “Diversity” can affect read depth

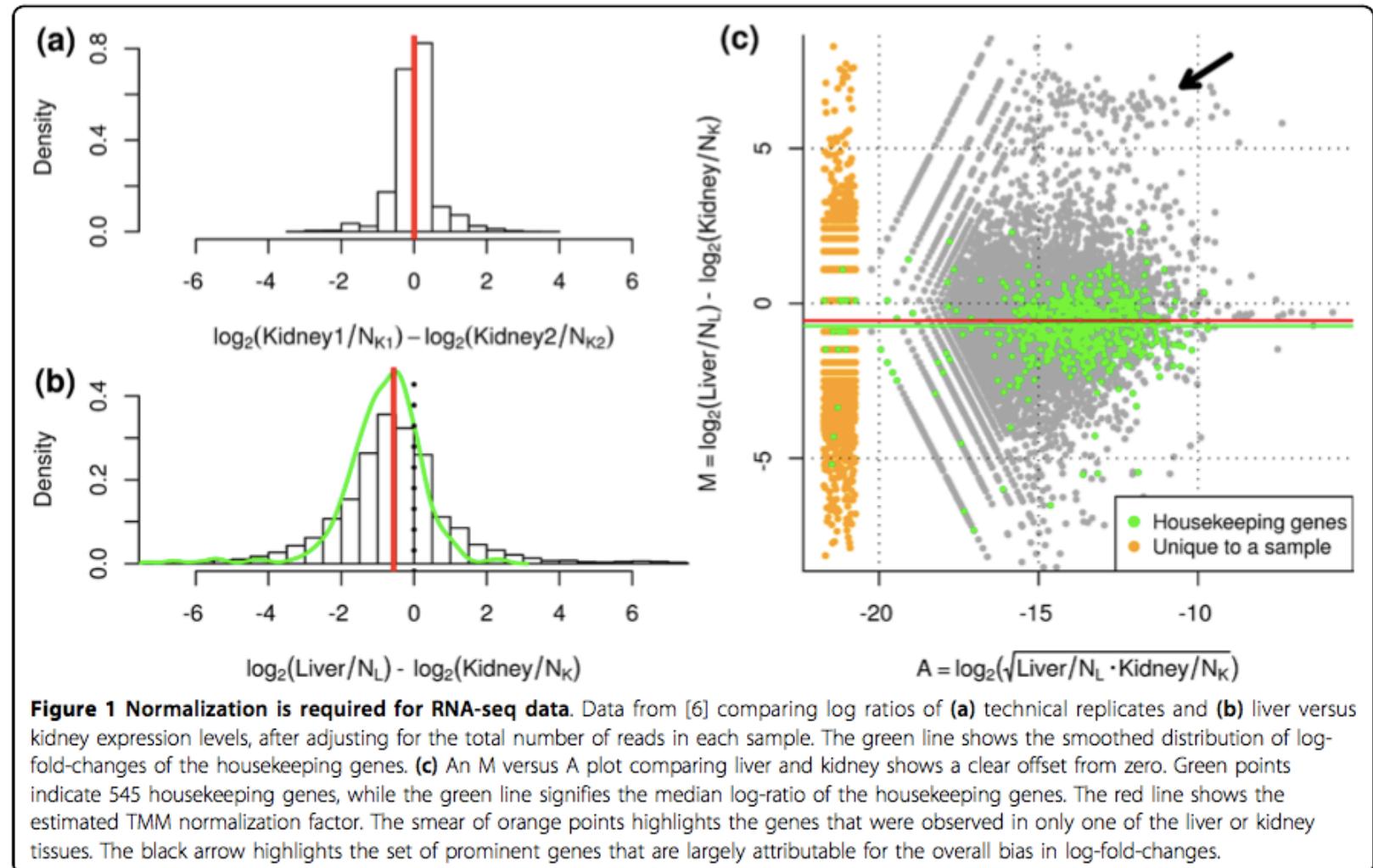
- Hypothetical example: Sequence 6 libraries to the same depth, with varying levels of unique-to-sample counts
- Read depth is affected not only by expression (and length), but also expression levels of other genes
- Composition can induce (sometimes significant) differences in counts

Red=low, goldenyellow=high





Kidney and Liver RNA have very different composition





Use scaling factor (“offset”) in statistical model

Assumption: core set of genes/loci that do not change in expression.

Our Pick a reference sample, compute a weighted trimmed mean of M-values (TMM) to reference

Adjustment to statistical analysis:

- Use “effective” library size (edgeR)
- Use additional offset (GLM)

Note: count data is not modified

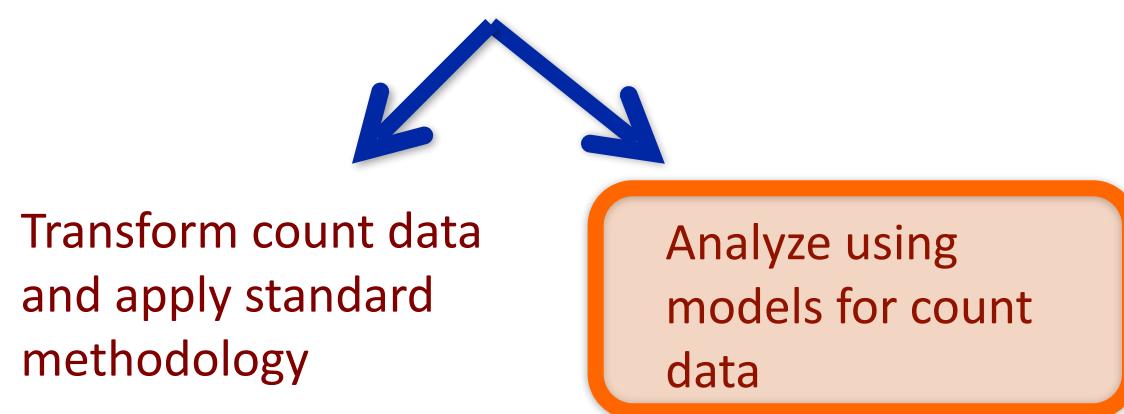
Differential expression: why not use methods developed for microarrays?

Count data is discrete, not continuous.

Methods designed for microarrays are not directly applicable and suboptimal

Transforming count data with logs, with some special treatment, can give very good results

Two options:





What does transformation do to M-V relationship?

For Poisson data, square-root should stabilize

Logarithm is too strong – variance decreases to asymptote (Neg Bin) or 0 (Poisson)

How to pick? Doesn't matter —> voom

voom: mean-variance modeling at the observational level

voom

package:limma

R Documentation

Transform RNA-Seq Data Ready for Linear Modelling

Description:

Transform count data to log2-counts per million, estimate the mean-variance relationship and use this to compute appropriate observational-level weights. The data are then ready for linear modeling.



Model log counts per million

log counts per million:

$$z_{gi} = \log_2 \left(1e6 \frac{\text{count}_{gi} + 0.5}{\text{libsize}_{gi} + 1.0} \right) = \log_2 \left(1e6 \frac{y_{gi} + 0.5}{M_{gi} + 1.0} \right)$$

normalize libsize in advance or normalize z_{gi} as for microarrays.

Linear modelling:

$$E(z_{gi}) = \mu_{gi} = x_i^T \beta_g$$

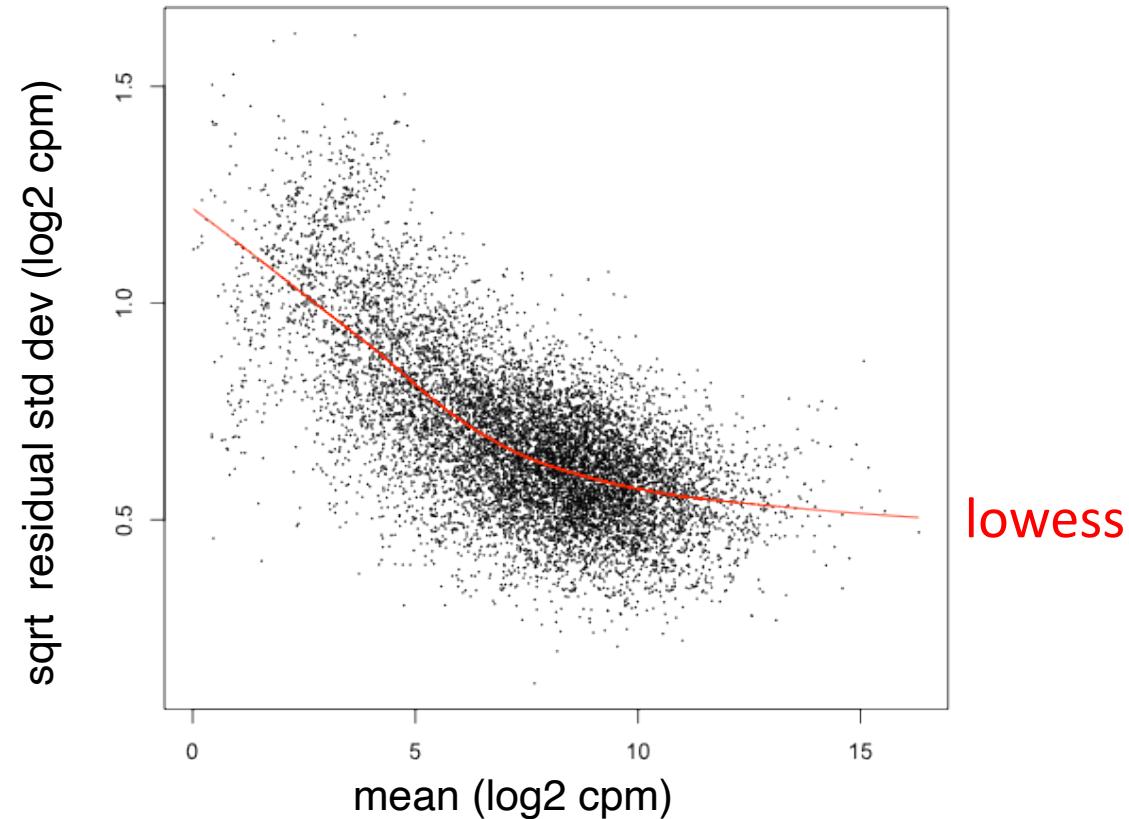
$$\text{var}(z_{gi}) = s(\mu_{gi}) \sigma_g^2$$



Smooth function of mean



voom fits a lowess trend to the mean-variance relationship ...



→ Use weights ($1/\text{var}$) in limma analysis .. i.e., **heteroscedastic regression**

