

(Source: O. Rueda, MRC-BSU; G. Marot, INRIA)

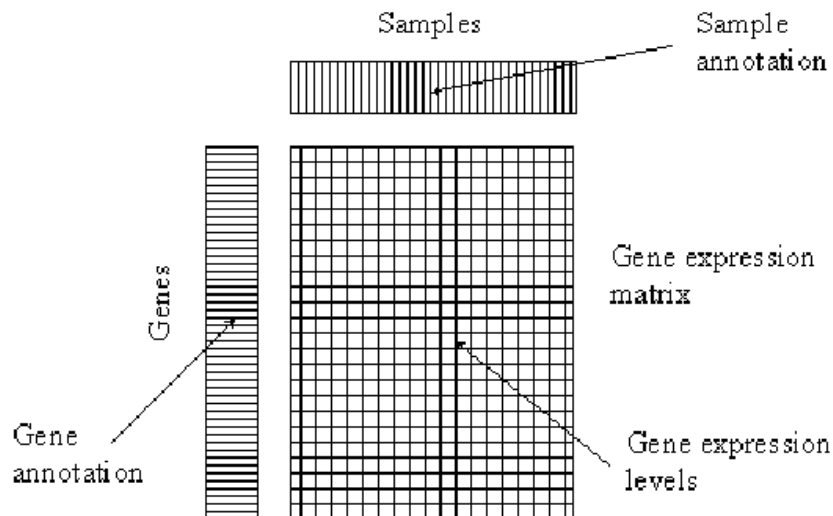
raw count for gene i , sample j

The mean is taken as "normalized counts" scaled by a normalization factor

one dispersion per gene

$$K_{ij} \sim \text{NB}(s_{ij}q_{ij}, \alpha_i)$$

Introduction



Introduction

```
> set.seed(777)
> cnts <- matrix(rnbinom(n=20000, mu=100, size=1/.25), ncol=20)
> cond <- factor(rep(1:2, each=10))

> dds <- DESeqDataSetFromMatrix(cnts, DataFrame(cond), ~ cond)
> dds <- DESeq(dds)
> results(dds)
```

log2 fold change (MLE): cond 2 vs 1

Wald test p-value: cond 2 vs 1

DataFrame with 1000 rows and 6 columns

	baseMean	log2FoldChange	lfcSE	stat	pvalue	padj
	<numeric>	<numeric>	<numeric>	<numeric>	<numeric>	<numeric>
1	97.3140	-0.682067	0.344525	-1.979730	0.0477339	0.745842
2	109.9860	-0.228819	0.450720	-0.507676	0.6116808	0.944354
3	98.8111	0.104291	0.462113	0.225683	0.8214483	0.978382
4	103.2615	0.306400	0.297682	1.029284	0.3033460	0.944354
5	97.9406	0.316338	0.357242	0.885501	0.3758864	0.944354
...
996	86.8057	0.0467703	0.287042	0.162939	0.8705668	0.980044
997	101.4437	-0.2070806	0.339886	-0.609264	0.5423495	0.944354
998	78.1356	-0.6372790	0.369515	-1.724637	0.0845930	0.824310
999	89.2920	0.7554725	0.306192	2.467314	0.0136131	0.614613
1000	103.5569	-0.0728875	0.348655	-0.209053	0.8344065	0.978382

Outline

▶ Part I: Quick recap

- ▷ Tests: Null and alternative hypotheses, Type I and type II errors, Power
- ▷ Experimental design & Sample size calculation.

▶ Part II: Modelling

- ▷ X design matrix,
- ▷ Linear regression,
- ▷ Negative binomial regression for counts.

▶ Part III: Multiplicity correction

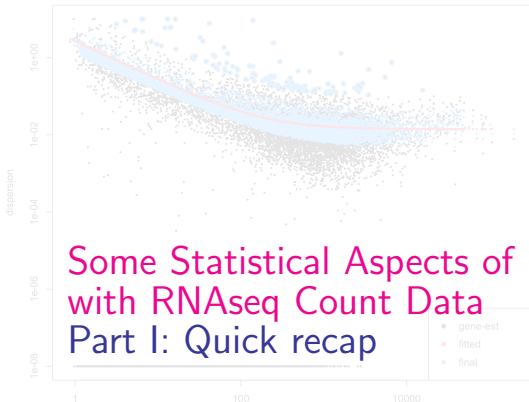
- ▷ Familywise error rate (FWER)
- ▷ False discovery rate (FDR)



The mean is taken as "normalized counts" scaled by a normalization factor

one dispersion per gene

$$K_{ij} \sim \text{NB}(s_{ij}q_{ij}, \alpha_i)$$



Some Statistical Aspects of DE Analysis with RNAseq Count Data Part I: Quick recap

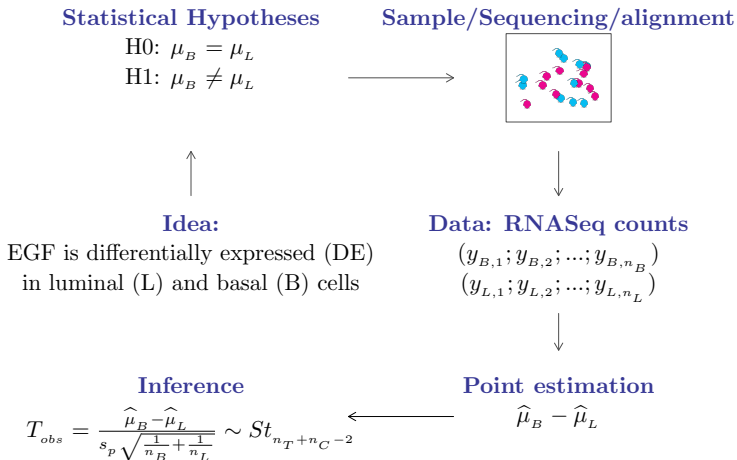
dominique-laurent.couturier@cruk.cam.ac.uk [Bioinformatics core]

The mean is taken as "normalized
counts" divided by a normalization
factor

one dispersion per gene

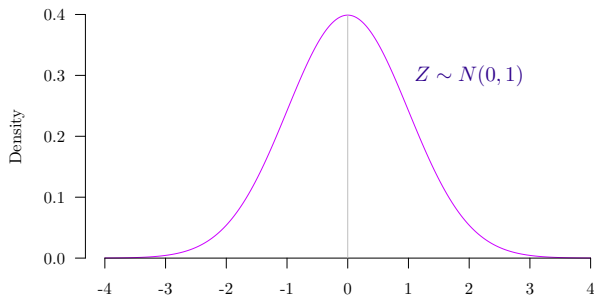
$$K_{ij} \sim \text{NB}(s_{ij}q_{ij}, \alpha_i)$$

Grand Picture of Statistics



Statistical tests

Assess how likely the observed test statistics is compared to the test statistics distribution under H_0 :



P-value for a two-sided test:

$$p\text{-value} = 2 \min [P(Z \leq Z_{obs} | H_0), P(Z \geq Z_{obs} | H_0)]$$

i.e. the probability of getting a test statistic as extreme or more extreme than the calculated test statistic if H_0 is true

Statistical tests

4 possible outcomes

Conclude:

- ▶ if $p\text{-value} > \alpha \rightarrow$ do not reject H_0 .
- ▶ if $p\text{-value} < \alpha \rightarrow$ reject H_0 in favour of H_1 .

		Test Outcome	
		H0 not rejected	H1 accepted
Unknown Truth	H0 true	$1 - \alpha$ [TN]	α [FP]
	H1 true	β [FN]	$1 - \beta$ [TP]

where

- ▶ α is the type I error, the probability of rejecting H_0 when H_0 is correct,
- ▶ β is the type II error, the probability of not rejecting H_0 when H_1 is correct.

Warnings

- ▶ 'absence of evidence is not evidence of absence',
- ▶ design may help minimising FP and FN (ie, maximising TN and TP).

Experimental design 1: Minimising biases

3 fundamental aspects of sounds experiments (Fisher 1935)

► Replication

Try to capture all sources of variability
(Biological versus technical variability)

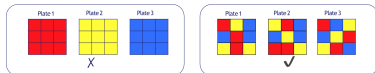
► Blocking

Try to remove technical biases/confounding
(Lane and batch effects)



► Randomisation

Try to remove confounding due to other factors



Experimental design 2: boosting power

Power- / Effect size- / Sample size- calculations

4 ingredients:

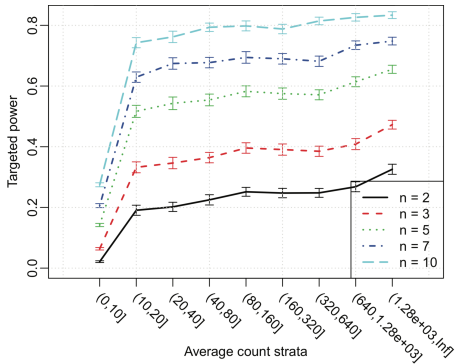
- ▶ $1 - \beta$, the power,
 - ▶ δ , the effect size: function of μ_L and μ_B
(log fold change, standardised difference),
 - ▶ n , the sample size (number of biological replicates),
 - ▶ α , the type I error.
- ▷ ϕ , nuisance parameters
(variability, sequencing depth, multiplicity correction)

'Give me 3 of them, I will deduce the fourth':

- ▶ **Power calculation:** Aim is to define the probability ($1 - \beta$) to detect an effect size of interest (δ) at the α level with a sample size of n biological replicates.
- ▶ **Sample size calculation:** Aim is to define the sample size (n) allowing to detect an effect size of interest (δ) at the α level with a given probability ($1 - \beta$).

Experimental design 2: boosting power

Power- calculations in DE analyses



(Wu, Wang and Wu (2015))

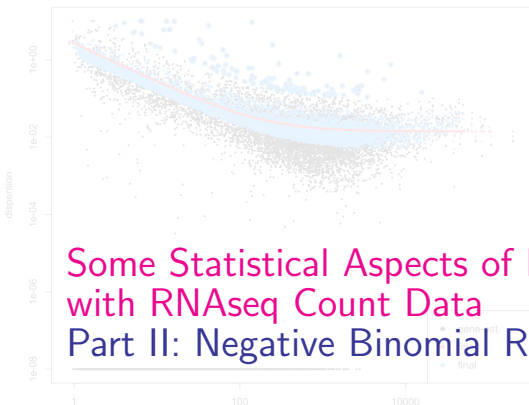


CANCER RESEARCH UK

CAMBRIDGE INSTITUTE



UNIVERSITY OF CAMBRIDGE



Some Statistical Aspects of DE Analysis with RNAseq Count Data

Part II: Negative Binomial Regression

dominique-laurent.couturier@cruc.cam.ac.uk [Bioinformatics core]

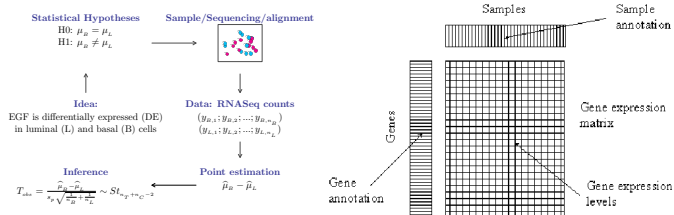
(Source: O. Rueda, MRC-BSU)

The mean is taken as "normalized count" multiplied by a normalization factor

one dispersion per gene

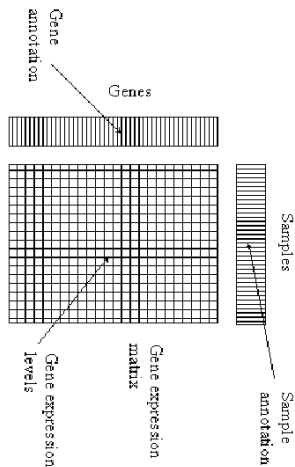
$$K_{ij} \sim \text{NB}(s_{ij}q_{ij}, \alpha_i)$$

Statistical modelling



Aim: Model the count data of each gene as a function of the conditions of interest (treatment, age, sex, batch, aso.)

Statistical modelling



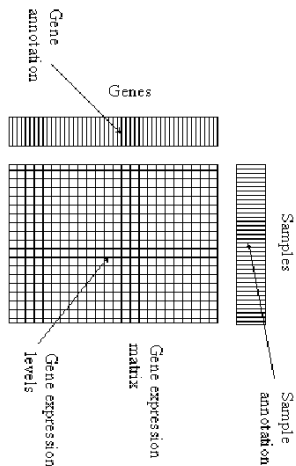
$$\mathbf{y} = f(\mathbf{X}) + \epsilon$$
$$E[\mathbf{y}] = f(\mathbf{X})$$

where

- ▶ \mathbf{y} denotes the $(n \times 1)$ vector of expression intensities of a given gene,
- ▶ \mathbf{X} denotes the $(n \times p)$ design/predictor matrix,
- ▶ ϵ denotes the $(n \times 1)$ stochastic error vector,
- ▶ $E[\mathbf{y}]$ denotes the expectation of \mathbf{y}

Express the count data vector of a given gene, \mathbf{y} , as a function f of characteristics of the samples (\mathbf{X} : age, treatment, aso) plus a stochastic error vector ϵ

Statistical modelling : Linear regression



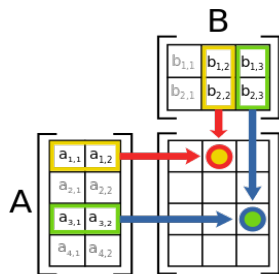
$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$$

$$\mathbb{E}[\mathbf{y}] = \mathbf{X}\boldsymbol{\beta}$$

where

- ▶ \mathbf{y} denotes the $(n \times 1)$ vector of expression intensities of a given gene,
- ▶ \mathbf{X} denotes the $(n \times p)$ design/predictor matrix,
- ▶ $\boldsymbol{\beta}$ denotes the $(p \times 1)$ parameter vector,
- ▶ $\boldsymbol{\epsilon} \sim N(0, \sigma^2)$ denotes the $(n \times 1)$ stochastic error vector,
- ▶ $\mathbb{E}[\mathbf{y}]$ denotes the expectation of \mathbf{y}

Statistical modelling : Linear regression



(Wikipedia)

$$y = X\beta + \epsilon$$
$$E[y] = X\beta$$

where

- ▶ y denotes the $(n \times 1)$ vector of expression intensities of a given gene,
- ▶ X denotes the $(n \times p)$ design/predictor matrix,
- ▶ β denotes the $(p \times 1)$ parameter vector,
- ▶ $\epsilon \sim N(0, \sigma^2)$ denotes the $(n \times 1)$ stochastic error vector,
- ▶ $E[y]$ denotes the expectation of y

Matrix multiplication:

the element $C_{i,j}$ (i th row, j th column of the matrix C) is obtained by

- ▶ multiplying **term-by-term** the entries of the i th row of A and the j th column of B ,
- ▶ and summing these products.

Statistical modelling : Strategy

- ▶ Collect the information related to each sample for the predictors of interest,
- ▶ define β , the sets of parameters we are interested in,
- ▶ build the \mathbf{X} matrix that relates the sample information with the β
this step is automatically done in R by specifying the regression formula in the function `lm()` or `DEseq2()`
- ▶ estimate the β and use statistical inference to assess significance (p -values)
these two points are done by the function `lm()` or `DEseq2()`

Statistical modelling : $\mathbf{X}\beta$ (For information)

- ▶ Linear regression:

$$E[\mathbf{y}] = \mathbf{X}\beta,$$

- ▶ Cox regression:

$$h(t) = h_0(t)e^{\mathbf{X}\beta},$$

- ▶ Logistic regression:

$$\pi = \frac{e^{\mathbf{X}\beta}}{1+e^{\mathbf{X}\beta}},$$

- ▶ Mean expression levels for a given gene in DESeq2:

$$E[\mathbf{y}] = 2^{\mathbf{X}\beta},$$

Statistical modelling : X contrast matrix

Contrast matrices for models with

- ▶ **one factor** / categorical predictor,
 - ▷ two experimental conditions (dichotomous predictor),
t-test
 - ▷ several experimental conditions,
One-way ANOVA
- ▶ **two factors** / categorical predictors,
 - ▷ without interaction,
 - ▷ with interaction,
Two-way ANOVA

Design matrix for models with a two-level factor

Sample	Treatment
Sample1	Treatment A
Sample 2	Control
Sample 3	Treatment A
Sample 4	Control
Sample 5	Treatment A
Sample 6	Control

Number of samples: 6

Number of factors: 1 with 2 levels (Control and Treatment A)

Possible parameters (What differences are important)?

- Effect of Treatment A
- Effect of Control

Design matrix for models with a two-level factor: No intercept

Sample	Treatment
Sample 1	Treatment A
Sample 2	Control
Sample 3	Treatment A
Sample 4	Control
Sample 5	Treatment A
Sample 6	Control

$$\begin{array}{l}
 \text{Sample 1} \\
 \text{Sample 2} \\
 \text{Sample 3} \\
 \text{Sample 4} \\
 \text{Sample 5} \\
 \text{Sample 6}
 \end{array}
 \begin{pmatrix} \\ \\ \\ \\ \\ \\ \end{pmatrix}
 =
 \begin{array}{l}
 \text{Control} \\
 \text{Treat. A} \\
 \\
 \\
 \\
 \\
 \end{array}
 \begin{pmatrix} \\ \\ \\ \\ \\ \\ \end{pmatrix}
 \beta
 \begin{bmatrix} \\ \beta_1 \\ \beta_2 \end{bmatrix}$$

y
 x

$\beta_1 = \mu_C$ is the mean expression of the control

$\beta_2 = \mu_A$ is the mean expression of the treatment A group

Design matrix for models with a two-level factor: With intercept

Sample	Treatment
Sample 1	Treatment A
Sample 2	Control
Sample 3	Treatment A
Sample 4	Control
Sample 5	Treatment A
Sample 6	Control

$$\begin{array}{l}
 \text{Sample 1} \\
 \text{Sample 2} \\
 \text{Sample 3} \\
 \text{Sample 4} \\
 \text{Sample 5} \\
 \text{Sample 6}
 \end{array}
 \begin{pmatrix} \\ \\ \\ \\ \\ \\ \end{pmatrix}
 =
 \begin{pmatrix}
 \text{intercept} \\
 \text{Shift bw C and A} \\
 \\ \\ \\ \\
 \end{pmatrix}
 \begin{bmatrix}
 \beta \\
 \beta_1 \\
 \beta_2
 \end{bmatrix}$$

y
 X

$\beta_1 = \mu_C$ is the mean expression of the control

β_2 is the shift in mean between the group A and the control group

Design matrices for models with a two-level factor: R Code

Open the R Markdown Document '[StatsRNAseq_Couturier.Rmd](#)'
and go to Section '[Contrast matrices / One 2-level factor](#)'

```
> dds <- DESeqDataSetFromMatrix(cnts, DataFrame(cond), ~ cond)
```

Design matrix for models with a three-level factor

Sample	Treatment
Sample1	Treatment A
Sample 2	Treatment B
Sample 3	Control
Sample 4	Treatment A
Sample 5	Treatment B
Sample 6	Control

Number of samples: 6

Number of factors: 1 with 3 levels (Control, Treatment A, Treatment B)

Possible parameters (What differences are important)?

- Effect of Treatment A
- Effect of Treatment B
- Effect of Control
- Differences between treatments?

Design matrix for models with a two-level factor: No intercept

Sample	Treatment
Sample1	Treatment A
Sample 2	Treatment B
Sample 3	Control
Sample 4	Treatment A
Sample 5	Treatment B
Sample 6	Control

$$\begin{bmatrix} \\ \\ \\ \\ \\ \end{bmatrix} = \begin{pmatrix} & \\ & \\ & \\ & \\ & \\ & \end{pmatrix} \begin{matrix} \beta \\ \beta_1 \\ \beta_2 \\ \beta_3 \end{matrix}$$

y x

$\beta_1 = \mu_C$ is the mean expression of the control

$\beta_2 = \mu_A$ is the mean expression of the treatment A group

$\beta_3 = \mu_B$ is the mean expression of the treatment B group

Design matrix for models with a two-level factor: With intercept

Sample	Treatment
Sample1	Treatment A
Sample 2	Treatment B
Sample 3	Control
Sample 4	Treatment A
Sample 5	Treatment B
Sample 6	Control

$$\begin{bmatrix} \\ \\ \\ \\ \\ \end{bmatrix} = \begin{pmatrix} & & \\ & & \\ & & \\ & & \\ & & \\ & & \end{pmatrix} \begin{matrix} \beta \\ \beta_1 \\ \beta_2 \\ \beta_3 \end{matrix}$$

y x

$\beta_1 = \mu_C$ is the mean expression of the control

β_2 is the shift in mean between the group A and the control group

β_3 is the shift in mean between the group B and the control group

Design matrices for models with a three-level factor: R Code

Open the R Markdown Document 'StatsRNAseq_Couturier.Rmd'
and go to Section 'Contrast matrices / One 3-level factor'

```
> dds <- DESeqDataSetFromMatrix(cnts, DataFrame(cond), ~ cond)
```

Design matrix for models with two two-level factors

Sample	Treatment	ER status
Sample1	Treatment A	+
Sample 2	No Treatment	+
Sample 3	Treatment A	+
Sample 4	No Treatment	+
Sample 5	Treatment A	-
Sample 6	No Treatment	-
Sample 7	Treatment A	-
Sample 8	No Treatment	-

Number of samples: 8

Number of factors: 2 two-level factors

Design matrix for models with two two-level factors: No interaction

Sample	Treatment	ER status
Sample1	Treatment A	+
Sample 2	No Treatment	+
Sample 3	Treatment A	+
Sample 4	No Treatment	+
Sample 5	Treatment A	-
Sample 6	No Treatment	-
Sample 7	Treatment A	-
Sample 8	No Treatment	-

$$\begin{bmatrix} \\ \\ \\ \\ \\ \\ \\ \end{bmatrix} = \begin{pmatrix} \\ \\ \\ \\ \\ \\ \\ \end{pmatrix} \begin{bmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \end{bmatrix}$$

Y X

$\beta_1 = \mu_C$ is the mean expression of the control

β_2 is the shift in mean between the group A and the control group

β_3 is the shift in mean between the ER+ group and the control group

Design matrix for models with two two-level factors: With interaction

Sample	Treatment	ER status
Sample 1	Treatment A	+
Sample 2	No Treatment	+
Sample 3	Treatment A	+
Sample 4	No Treatment	+
Sample 5	Treatment A	-
Sample 6	No Treatment	-
Sample 7	Treatment A	-
Sample 8	No Treatment	-

$$\begin{bmatrix} \\ \\ \\ \\ \\ \\ \\ \end{bmatrix} = \begin{pmatrix} \\ \\ \\ \\ \\ \\ \\ \end{pmatrix} \begin{bmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \\ \beta_4 \end{bmatrix}$$

Y
 X

$\beta_1 = \mu_C$ is the mean expression of the control

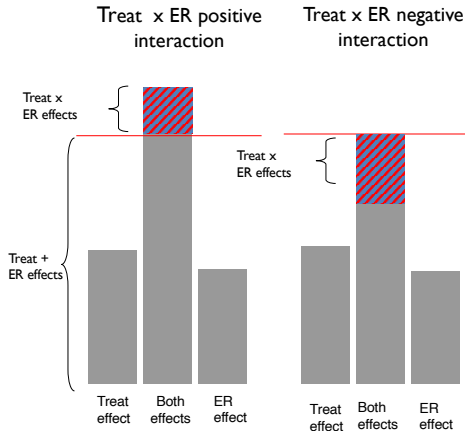
β_2 is the shift in mean between the group A and the control group

β_3 is the shift in mean between the ER+ group and the control group

β_4 is the additional shift in mean for patients of the ER+ and Treatment A groups

Design matrix for models with two two-level factors: With interaction

Sample	Treatment	ER status
Sample1	Treatment A	+
Sample 2	No Treatment	+
Sample 3	Treatment A	+
Sample 4	No Treatment	+
Sample 5	Treatment A	-
Sample 6	No Treatment	-
Sample 7	Treatment A	-
Sample 8	No Treatment	-



Design matrices for models with two two-level factors: R Code

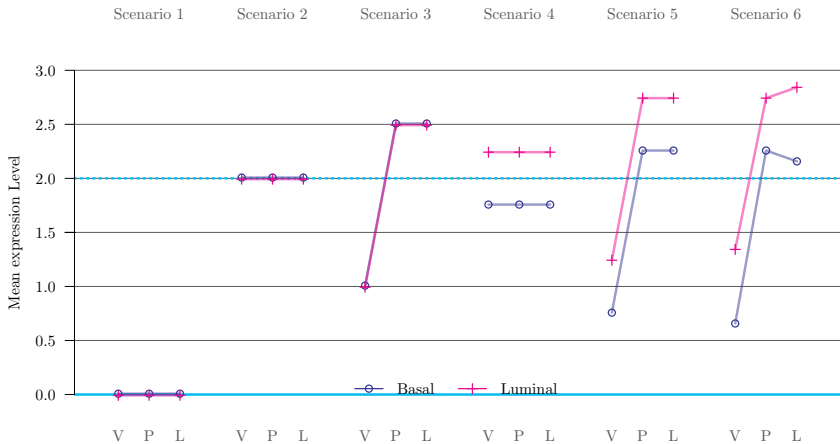
Open the R Markdown Document 'StatsRNAseq_Couturier.Rmd'
and go to Section 'Contrast matrices / Two 2-level factors'

```
> dds <- DESeqDataSetFromMatrix(cnts, DataFrame(cond), ~ cond)
```

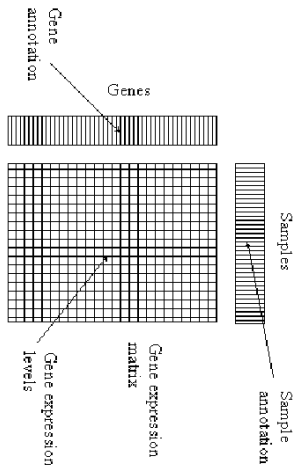

Models with 2 factors: possible scenarios

2 factors:

- ▶ cell type (2 levels): luminal versus basal
- ▶ mouse type (3 levels): virgin, pregnant, lactating



Negative binomial regression: Model



$$y \sim \text{NB}(\mu, \phi)$$

$$E[y] = \mu = s 2^{\mathbf{X}\beta}$$

where

- ▶ y denotes the $(n \times 1)$ **count vector** of expression intensities of a given gene,
- ▶ \mathbf{X} denotes the $(n \times p)$ **design/predictor matrix**,
- ▶ β denotes the $(p \times 1)$ **parameter vector**,
- ▶ ϕ denotes the **dispersion parameter**,
- ▶ s denotes the **scaling factor vector** (library size),
- ▶ $E[y] = \mu$ denotes the expectation of y

Negative binomial regression:

Probability mass function

$$y \sim \text{NB}(\mu, \phi)$$

$$f(y|\mu, \phi) = \frac{\Gamma(y + \frac{1}{\phi})}{\Gamma(\frac{1}{\phi})\Gamma(y + 1)} \left(\frac{\phi\mu}{1 + \phi\mu} \right)^y \left(\frac{1}{1 + \phi\mu} \right)^{\frac{1}{\phi}}$$

with expectation and variance given by

► $E[y] = \mu = \mathbf{s} \mathbf{2}^{\mathbf{X}\beta}$

► $\text{Var}[y] = \mu \left(1 + \frac{\mu}{\phi} \right)$

Negative binomial regression: Log2 FC

log2 fold change (MLE): cond 2 vs 1

Wald test p-value: cond 2 vs 1

DataFrame with 1000 rows and 6 columns

	baseMean	log2FoldChange	lfcSE	stat	pvalue	padj
	<numeric>	<numeric>	<numeric>	<numeric>	<numeric>	<numeric>
1	97.3140	-0.682067	0.344525	-1.979730	0.0477339	0.745842
2	109.9860	-0.228819	0.450720	-0.507676	0.6116808	0.944354
...
999	89.2920	0.7554725	0.306192	2.467314	0.0136131	0.614613
1000	103.5569	-0.0728875	0.348655	-0.209053	0.8344065	0.978382

► $E[\mathbf{y} | \text{'cond 1'}] = 2^{\hat{\beta}_1}$

► $E[\mathbf{y} | \text{'cond 2'}] = 2^{\hat{\beta}_1 + \hat{\beta}_2} = 2^{\hat{\beta}_1} 2^{\hat{\beta}_2}$

▷ If not DE, $\beta_2 = 0$ so that $E[\mathbf{y} | \text{'cond 2'}] = 2^{\hat{\beta}_1} 2^0 = 2^{\hat{\beta}_1}$,

▷ If DE, $\beta_2 \neq 0$ so that $E[\mathbf{y} | \text{'cond 2'}] = 2^{\hat{\beta}_1} 2^{\hat{\beta}_2}$

Interpretation: *Multiplicative change in observed gene expression level of $2^{\hat{\beta}_2} = 2^{-0.682067} = 0.6232717$ compared to the condition 1*

Negative binomial regression: Significance

```
log2 fold change (MLE): cond 2 vs 1
Wald test p-value: cond 2 vs 1
DataFrame with 1000 rows and 6 columns
  baseMean log2FoldChange lfcSE      stat      pvalue      padj
  <numeric> <numeric> <numeric> <numeric> <numeric> <numeric>
1      97.3140      -0.682067  0.344525 -1.979730 0.0477339 0.745842
2     109.9860      -0.228819  0.450720 -0.507676 0.6116808 0.944354
...
999    89.2920      0.7554725  0.306192  2.467314 0.0136131 0.614613
1000  103.5569     -0.0728875  0.348655 -0.209053 0.8344065 0.978382
```

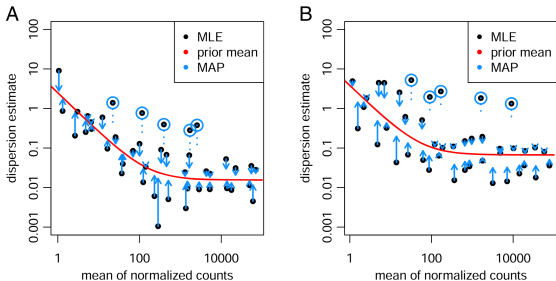
Wald Z-test to assess if a Log2 FC is significantly different from 0:

- ▶ **H0:** $\beta_2 = 0$ versus **H1:** $\beta_2 \neq 0$
- ▶ Z-statistic = $\frac{\hat{\beta}_2}{\hat{\sigma}_{\hat{\beta}_2}} = \frac{-0.682067}{0.344525} = -1.979730$
- ▶ P-value with $Z \sim N(0, 1)$ under **H0** is given by
> 2*(1-pnorm(abs(-1.979730)))

```
[1] 0.04773388
```

Negative binomial regression: Assumed Distribution

- ▶ The **assumed distribution of counts per condition for a given gene** depends on
 - ▷ $\hat{\beta}$, the estimate of the parameter vector,
 - ▷ $\hat{\phi}$, the estimate of the dispersion parameter for that gene.
- ▶ There are **3 ways to estimate ϕ in DESeq2**:
 - ▷ **gene-wise** dispersion estimates via ML (black dots) [not efficient],
 - ▷ **smooth curve** (red line) [strong assumption],
 - ▷ Bayesian **combination of both** [mid-way optimal solution].



Negative binomial regression: Assumed Distribution

```
-> mcols(dds)[,c("Intercept", "cond_2_vs_1", "dispGeneEst", "dispFit", "dispersion")]
```

```
DataFrame with 1000 rows and 5 columns
```

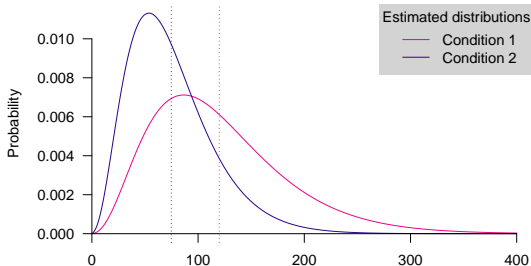
	Intercept	cond_2_vs_1	dispGeneEst	dispFit	dispersion
	<numeric>	<numeric>	<numeric>	<numeric>	<numeric>
1	6.90565	-0.682067	0.294082	0.234624	0.274708
2	6.89102	-0.228819	0.479231	0.230525	0.479231
...
999	6.05380	0.7554725	0.206644	0.229562	0.213730
1000	6.73029	-0.0728875	0.304930	0.235483	0.282745

- For gene 1 and condition 1, we have

$$y \sim \text{NB}(\hat{\mu} = 2^{6.90565} = 119.8969, \hat{\phi} = 0.274708)$$

- For gene 1 and condition 2, we have

$$y \sim \text{NB}(\hat{\mu} = 2^{6.90565} 2^{-0.682067} = 74.72831, \hat{\phi} = 0.274708)$$



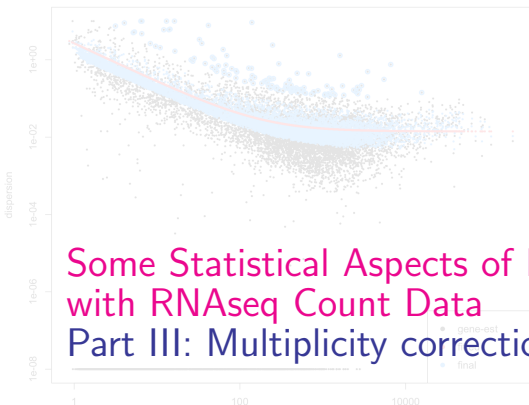


CANCER RESEARCH UK

CAMBRIDGE INSTITUTE



UNIVERSITY OF CAMBRIDGE

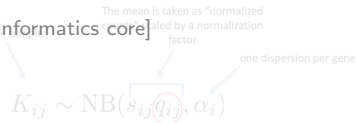


Some Statistical Aspects of DE Analysis with RNAseq Count Data

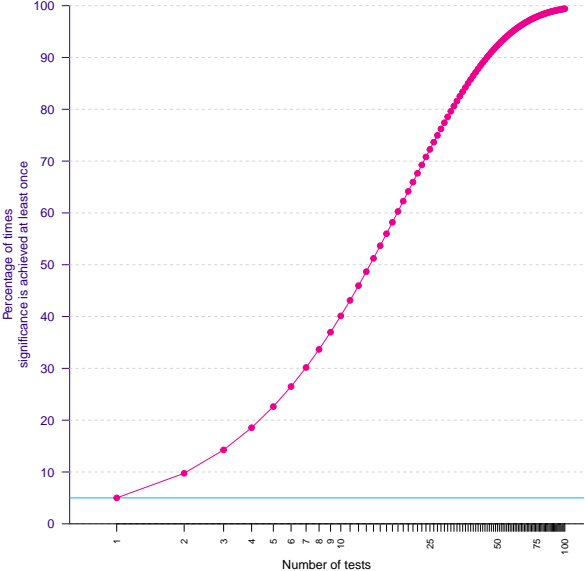
Part III: Multiplicity correction

dominique-laurent.couturier@cruc.cam.ac.uk [Bioinformatics core]

(Source: G. Marot, INRIA)



Multiplicity correction: Familywise error rate



Multiplicity correction

Experimental design

Exploration

Normalization

Differential analysis

Multiple testing

The Family Wise Error Rate (FWER)

Definition

Probability of having at least one Type I error (false positive), of declaring DE at least one non DE gene.

$$FWER = \mathbb{P}(FP \leq 1)$$

The Bonferroni procedure

Either each test is realized at $\alpha = \alpha^*/G$ level
or use of adjusted pvalue $pBonf_i = \min(1, p_i * G)$ and $FWER \leq \alpha^*$.
For $G = 2000$, $\leq \alpha^* = 0.05$, $\alpha = 2.510^{-5}$.

Easy but conservative and not powerful.

Multiplicity correction

Experimental design

Exploration

Normalization

Differential analysis

Multiple testing

The False Discovery Rate (FDR)

Idea : Do not control the error rate but the proportion of error
⇒ less conservative than control of the FWER.

Definition

The false discovery rate of [Benjamini and Hochberg, 1995] is the expected proportion of Type I errors among the rejected hypotheses

$$\text{FDR} = \mathbb{E}(FP/P) \text{ if } P > 0 \text{ and } 0 \text{ if } P = 0$$

Prop

$$\text{FDR} \leq \text{FWER}$$

Multiplicity correction

```
> set.seed(777)
> cnts <- matrix(rnbinom(n=20000, mu=100, size=1/.25), ncol=20)
> cond <- factor(rep(1:2, each=10))

> dds <- DESeqDataSetFromMatrix(cnts, DataFrame(cond), ~ cond)
> dds <- DESeq(dds)
> results(dds)
```

log₂ fold change (MLE): cond 2 vs 1

Wald test p-value: cond 2 vs 1

DataFrame with 1000 rows and 6 columns

	baseMean	log2FoldChange	lfcSE	stat	pvalue	padj
	<numeric>	<numeric>	<numeric>	<numeric>	<numeric>	<numeric>
1	97.3140	-0.682067	0.344525	-1.979730	0.0477339	0.745842
2	109.9860	-0.228819	0.450720	-0.507676	0.6116808	0.944354
3	98.8111	0.104291	0.462113	0.225683	0.8214483	0.978382
4	103.2615	0.306400	0.297682	1.029284	0.3033460	0.944354
5	97.9406	0.316338	0.357242	0.885501	0.3758864	0.944354
...
996	86.8057	0.0467703	0.287042	0.162939	0.8705668	0.980044
997	101.4437	-0.2070806	0.339886	-0.609264	0.5423495	0.944354
998	78.1356	-0.6372790	0.369515	-1.724637	0.0845930	0.824310
999	89.2920	0.7554725	0.306192	2.467314	0.0136131	0.614613
1000	103.5569	-0.0728875	0.348655	-0.209053	0.8344065	0.978382

```
> p.adjust(results(dds)[, "pvalue"], method="BH")[c(1:5, 996:1000)]
```

```
[1] 0.7458417 0.9443538 0.9783822 0.9443538 0.9443538 0.9800445 0.9443538 0.8243099
[9] 0.6146133 0.9783822
```

Multiplicity correction

Experimental design

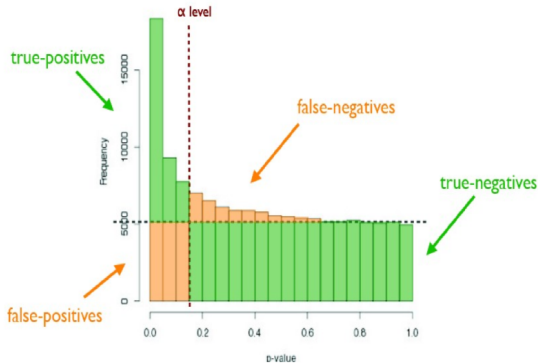
Exploration

Normalization

Differential analysis

Multiple testing

Standard assumption for p-value distribution



Source : M. Guedj, Pharnext

Multiplicity correction

Experimental design

Exploration

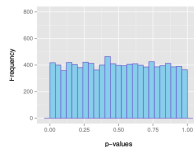
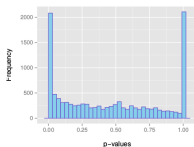
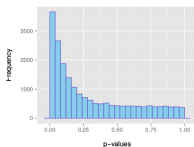
Normalization

Differential analysis

Multiple testing

p-values histograms for diagnosis

Examples of **expected overall distribution**



(a) : the most desirable shape

(b) : very low counts genes usually have large p-values

(c) : do not expect positive tests after correction

Multiplicity correction

Experimental design

Exploration

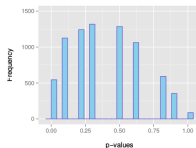
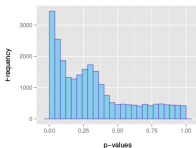
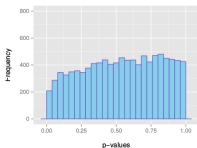
Normalization

Differential analysis

Multiple testing

p-values histograms for diagnosis

Examples of not expected overall distribution



- (a) : indicates a batch effect (confounding hidden variables)
- (b) : the test statistics may be inappropriate (due to strong correlation structure for instance)
- (c) : discrete distribution of p-values : unexpected

CONCLUSION

```
> set.seed(777)
> cnts <- matrix(rnbinom(n=20000, mu=100, size=1/.25), ncol=20)
> cond <- factor(rep(1:2, each=10))

> dds <- DESeqDataSetFromMatrix(cnts, DataFrame(cond), ~ cond)
> dds <- DESeq(dds)
> results(dds)
```

log2 fold change (MLE): cond 2 vs 1

Wald test p-value: cond 2 vs 1

DataFrame with 1000 rows and 6 columns

	baseMean	log2FoldChange	lfcSE	stat	pvalue	padj
	<numeric>	<numeric>	<numeric>	<numeric>	<numeric>	<numeric>
1	97.3140	-0.682067	0.344525	-1.979730	0.0477339	0.745842
2	109.9860	-0.228819	0.450720	-0.507676	0.6116808	0.944354
3	98.8111	0.104291	0.462113	0.225683	0.8214483	0.978382
4	103.2615	0.306400	0.297682	1.029284	0.3033460	0.944354
5	97.9406	0.316338	0.357242	0.885501	0.3758864	0.944354
...
996	86.8057	0.0467703	0.287042	0.162939	0.8705668	0.980044
997	101.4437	-0.2070806	0.339886	-0.609264	0.5423495	0.944354
998	78.1356	-0.6372790	0.369515	-1.724637	0.0845930	0.824310
999	89.2920	0.7554725	0.306192	2.467314	0.0136131	0.614613
1000	103.5569	-0.0728875	0.348655	-0.209053	0.8344065	0.978382