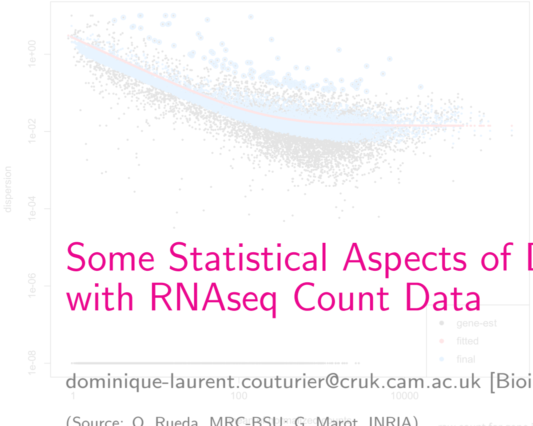
CANCER RESEARCH UK | CAMBRIDGE INSTITUTE

UNIVERSITY OF CAMBRIDGE



Some Statistical Aspects of DE Analysis with RNAseq Count Data

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

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

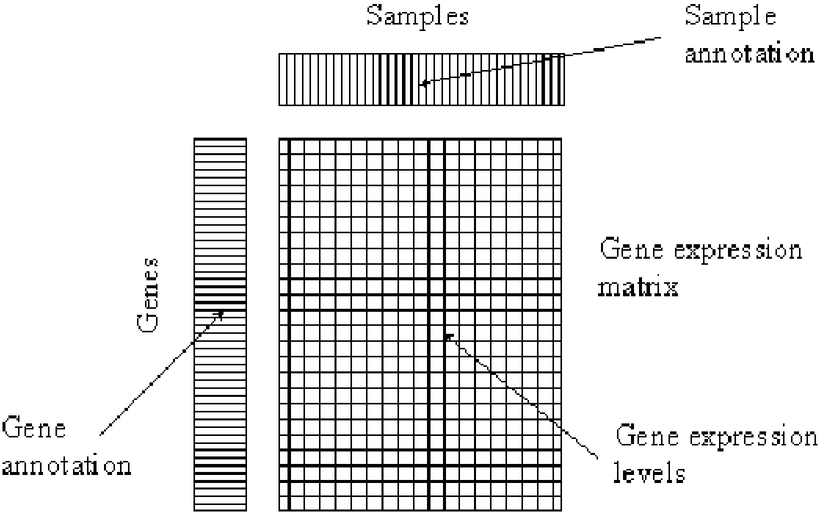
raw count for gene i, sample j

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

one dispersion per gene

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

Introduction



Introduction

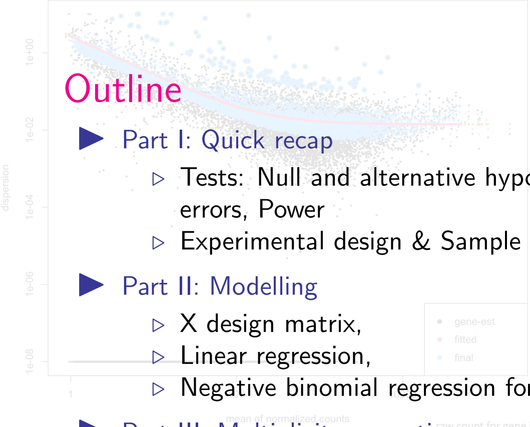
```
> set.seed(777)
> cnts <- matrix(rnbinom(n=20000, mu=100, size=1/.25), ncol=20, nrow=1000)
> cond <- factor(rep(1:2, each=10))

> 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)

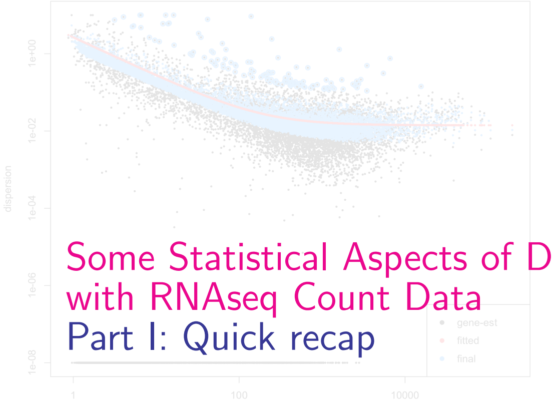


raw count for gene i, sample j

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

one dispersion per gene

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



Some Statistical Aspects of DE Analysis with RNAseq Count Data

Part I: Quick recap

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

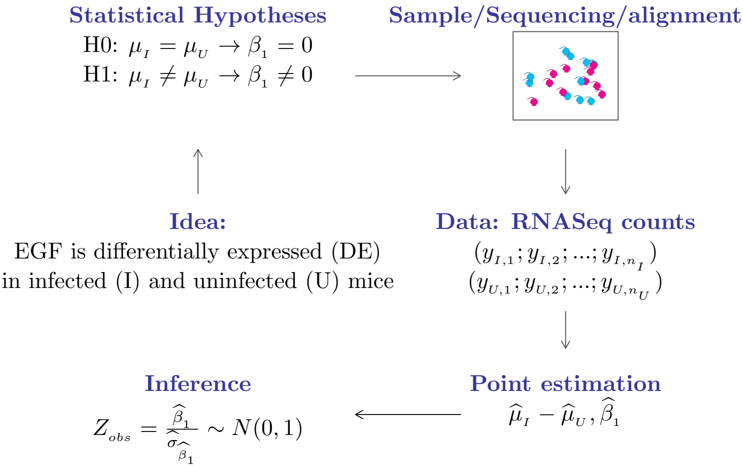
raw count for gene i, sample j

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

one dispersion per gene

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

Grand Picture of Statistics



Idea:

EGF is differentially expressed (DE)
in infected (I) and uninfected (U) mice

Statistical Hypotheses

H0: $\mu_I = \mu_U \rightarrow \beta_1 = 0$

H1: $\mu_I \neq \mu_U \rightarrow \beta_1 \neq 0$

Sample/Sequencing/alignment

Data: RNASeq counts

$(y_{I,1}; y_{I,2}; \ldots; y_{I,n_I})$

$(y_{U,1}; y_{U,2}; \ldots; y_{U,n_U})$

Point estimation

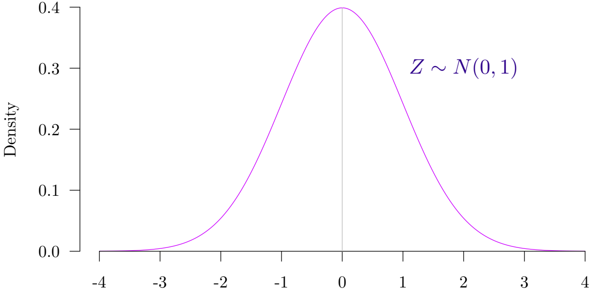
$\widehat{\mu}_I - \widehat{\mu}_U, \widehat{\beta}_1$

Inference

$$Z_{obs} = \frac{\widehat{\beta}_1}{\widehat{\sigma}_{\widehat{\beta}_1}} \sim N(0,1)$$

Statistical tests

Compare the observed test statistics, Z_{obs}, to its distribution under H0 to assess how likely it is to observe such a value if there is no effect:



P-value for a two-sided test:
p-value $= 2 \min \left[P(Z \leq Z_{obs}|\text{H0}), P(Z \geq Z_{obs}|\text{H0})\right]$
i.e. the probability of getting a test statistic as extreme or more extreme than the calculated test statistic if H0 is true

Statistical tests

4 possible outcomes

Conclude:

- if p-value $> \alpha$ → do not reject H0.
- if p-value $< \alpha$ → reject H0 in favour of H1.

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

where

- α is the type I error, the probability of rejecting H0 when H0 is correct,
- θ is the type II error, the probability of not rejecting H0 when H1 is correct.

Warnings

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

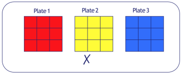
Experimental design 1: Minimising biases

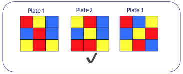
3 fundamental aspects of sounds experiments (Fisher 1935)

- 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:

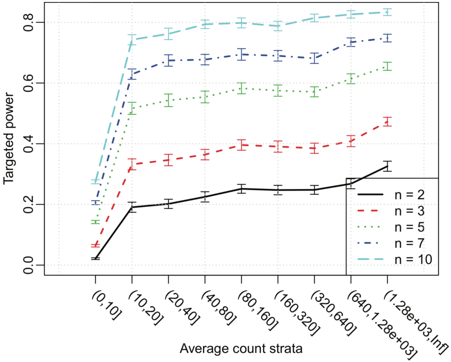
- $1-\theta$, the power,
- δ, the effect size: function of μ_{U} and μ_{I} (log fold change, standardised difference),
- n, the sample size (number of biological replicates),
- α, the type I error.
- ϕ, nuisance parameters (variability, sequencing depth, multiplicity correction)

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

- Power calculation: Aim is to define the probability ($1-\theta$) 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 ($1-\theta$).

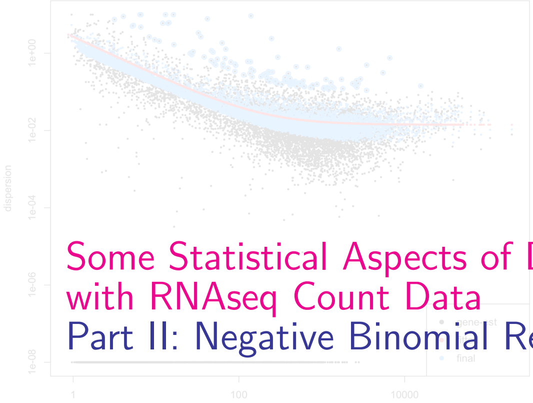
Experimental design 2: boosting power

Power- calculations in DE analyses



(Wu, Wang and Wu (2015))

Coffee break



Some Statistical Aspects of DE Analysis with RNAseq Count Data

Part II: Negative Binomial Regression

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

(Source: O. Rueda, MRC-BSU)

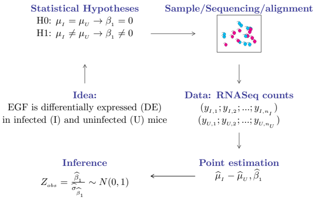
raw count for gene i, sample j

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

one dispersion per gene

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

Statistical modelling



Statistical Hypotheses

H0: $\mu_I = \mu_U \rightarrow \beta_1 = 0$

H1: $\mu_I \neq \mu_U \rightarrow \beta_1 \neq 0$

Sample/Sequencing/alignment

Idea:
EGF is differentially expressed (DE)
in infected (I) and uninfected (U) mice

Data: RNASeq counts

$(y_{I,1}; y_{I,2}; ...; y_{I,n_I})$

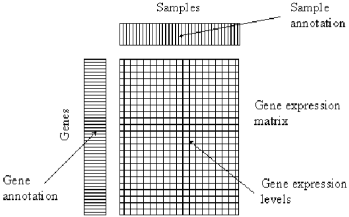
$(y_{U,1}; y_{U,2}; ...; y_{U,n_U})$

Inference

$Z_{obs} = \frac{\widehat{\beta}_1}{\widehat{\sigma}_{\widehat{\beta}_1}} \sim N(0,1)$

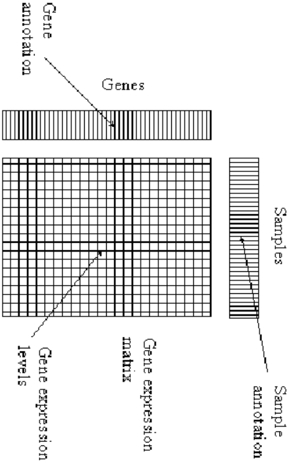
Point estimation

$\widehat{\mu}_I - \widehat{\mu}_U, \widehat{\beta}_1$



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

Statistical modelling



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

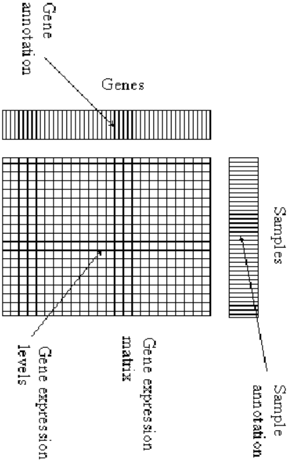
$$\mathsf{E}[\mathbf{y}] = f(\mathbf{X})$$

where

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



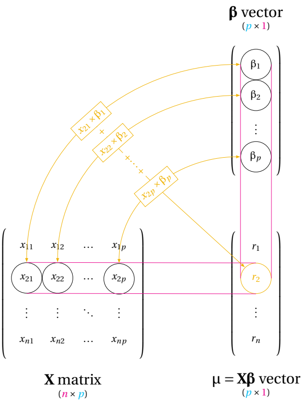
$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \epsilon$$

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

where

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

Statistical modelling : Linear regression



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

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

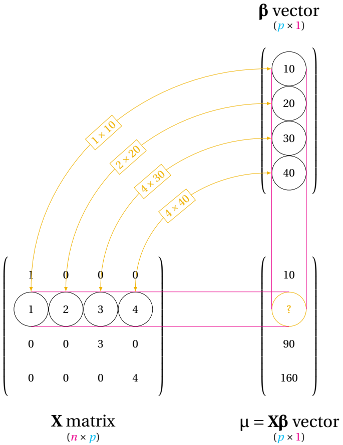
where

- $\mathbf{y}$ 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,
- $\epsilon \sim N(0, \sigma^2)$ denotes the (n $\times$ 1) stochastic error vector,
- $\mathsf{E}[\mathbf{y}]$ denotes the expectation of $\mathbf{y}$

Matrix multiplication:
the ith element $\boldsymbol{\mu} = \mathbf{X}\boldsymbol{\beta}$ is obtained by

- multiplying **term-by-term** the entries of the ith row of $\mathbf{X}$ and each element of β,
- and summing these products.

Statistical modelling : Linear regression



Matrix multiplication:
the ith element $\mathbf{r} = \mathbf{X}\beta$ is obtained by

- multiplying **term-by-term** the entries of the ith row of **X** and each element of β,
- and summing these products.

Statistical modelling : Strategy

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

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

- Linear regression:
  $E[\mathbf{y}] = \mathbf{X}\boldsymbol{\beta}$,

- Cox regression:
  $h(t) = h_0(t)e^{\mathbf{X}\boldsymbol{\beta}}$,

- Logistic regression:
  $\boldsymbol{\pi} = \frac{e^{\mathbf{X}\boldsymbol{\beta}}}{1+e^{\mathbf{X}\boldsymbol{\beta}}}$,

- Mean expression levels for a given gene in DESeq2:
  $E[\mathbf{y}] = 2^{\mathbf{X}\boldsymbol{\beta}}$,

Statistical modelling : X contrast matrix

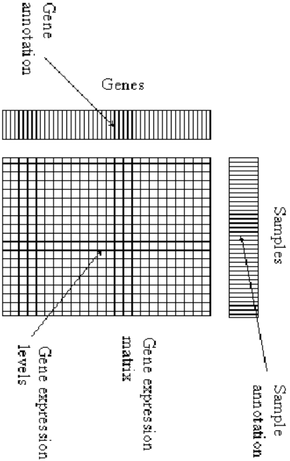
We will discuss contrast matrices for models with

- 1 factor (1 categorical predictor),
  - 2 experimental conditions
    (binary predictor: control/treatment),
    t-test
  - >2 experimental conditions
    (categorical predictor, like control/treatment 1/treatment 2),
    One-way ANOVA

- 2 factors (2 categorical predictors),
  - without interaction,
  - with interaction,

  Two-way ANOVA

Example: Toxoplasma Gondii Oocysts



#	Sample ID	Status	Time Point
1	SRR7657878	Infected	11 dpi
2	SRR7657881	Infected	11 dpi
3	SRR7657880	Infected	11 dpi
4	SRR7657874	Infected	33 dpi
5	SRR7657882	Uninfected	33 dpi
6	SRR7657872	Infected	33 dpi
7	SRR7657877	Uninfected	11 dpi
8	SRR7657876	Uninfected	11 dpi
9	SRR7657879	Uninfected	11 dpi
10	SRR7657883	Uninfected	33 dpi
11	SRR7657873	Infected	33 dpi
12	SRR7657875	Uninfected	33 dpi

2 Factors:

- Status with 2 levels (Infected/uninfected)
- Time point with 2 levels (11 dpi, 13 dpi)

https://github.com/bioinformatics-core-shared-training/Bulk_RNAseq_Course_2021/blob/main/Course_Materials/data/samplesheet_corrected.tsv

Case 1: 1 two-level factor without intercept

Modelling 1:

- Mean expression level of gene 'G' is a function of Status: Uninfected and infected.
- 2 levels = 2 parameters

$$\begin{pmatrix} \mu_U \\ \mu_I \end{pmatrix}$$

β vector

Sample information
(1 two-level factor)
I for 'Infected', U for 'Uninfected'

$$\begin{array}{c} I\\ I\\ I\\ I\\ U\\ I\\ U\\ U\\ U\\ U\\ I\\ U \end{array} \quad \begin{pmatrix} \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \end{pmatrix} \quad \begin{pmatrix} \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \end{pmatrix}$$

X matrix
(12×2)

Xβ vector
$(p \times 1)$

Parameters: $\boldsymbol{\beta} = [\mu_U, \mu_I]^T$, where

- μ_U denoted the mean expression level for condition 'Uninfected'
- μ_I denoted the mean expression level for condition 'Infected'

Case 2: 1 two-level factor with intercept

Modelling 2:

- Mean expression level of gene 'G' is a function of Status: Uninfected and infected.
- 2 levels = 2 parameters

Parameters: $\boldsymbol{\beta} = [\beta_0, \beta_1]^T$, where

- $\beta_0 = \mu_{\text{U}}$ is the intercept and corresponds to the mean expression level for the reference group: condition 'Uninfected'.
- $\beta_1 = \mu_{\text{I}} - \mu_{\text{U}}$ is the difference in mean expression level between conditions 'Infected' and 'Uninfected'

$$\begin{pmatrix} \beta_0 \\ \beta_1 \end{pmatrix}$$

β vector

Sample information
(1 two-level factor)
I for 'Infected', U for 'Uninfected'

$$\begin{array}{c} \text{I} \\ \text{I} \\ \text{I} \\ \text{I} \\ \text{U} \\ \text{I} \\ \text{U} \\ \text{U} \\ \text{U} \\ \text{U} \\ \text{I} \\ \text{U} \end{array} \qquad \begin{pmatrix} \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \end{pmatrix} \qquad \begin{pmatrix} \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \end{pmatrix}$$

X matrix
(12×2)

Xβ vector
$(p \times 1)$

Design matrices for models with one factor: R Code

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

Open the R Markdown Document 'StatsRNAseq_Couturier.Rmd' and go to Sections 'Contrast matrices / One 2-level factor' and 'Contrast matrices / One 3-level factor'

Case 5: 2 two-level factors without interaction

Modelling 1:

- Mean expression level of gene 'G' is a function of Status (Uninfected and infected) and Time (11 and 33 dpi).
- 2 (Status levels) × 2 (Time levels) = 3 parameters without interaction

Parameters: $\boldsymbol{\beta} = [\beta_0, \beta_1, \beta_2]^T$, where

- $\beta_0 = \mu_{U,11}$ denoted the mean expression level for the reference group: condition 'Uninfected' at 'Time 11'
- β_1 denoted the shift in mean due to condition 'Infected'
- β_2 denoted the shift in mean due to condition 'Time 33'

$$\begin{pmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{pmatrix}$$

β vector

Sample information
(2 two-level factors)
I for 'Infected', U for 'Uninfected'
11 for '11 dpi' and 33 for '33 dpi'

I	11
I	11
I	11
I	33
U	33
I	33
U	11
U	11
U	11
U	33
I	33
U	33

$$\begin{pmatrix} \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \end{pmatrix} \qquad \begin{pmatrix} \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \end{pmatrix}$$

X matrix
(12×3)

Xβ vector
$(p \times 1)$

Case 5: 2 two-level factors with interaction

Modelling 1:

- Mean expression level of gene 'G' is a function of Status (Uninfected and infected) and Time (11 and 33 dpi).
- 2 (Status levels) × 2 (Time levels) = 4 parameters with interaction

Parameters: $\boldsymbol{\beta} = [\beta_0, \beta_1, \beta_2, \beta_3]^T$, where

- $\beta_0 = \mu_{U,11}$ denoted the mean expression level for the reference group: condition 'Uninfected' at 'Time 11'
- β_1 denoted the shift in mean due to condition 'Infected'
- β_2 denoted the shift in mean due to condition 'Time 33'
- β_3 denoted the shift in mean due to conditions 'Infected' & 'Time 33' jointly given the main effects of 'Status' and 'Time'

$$\begin{pmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \beta_3 \end{pmatrix}$$

β vector

Sample information
(2 two-level factors)
I for 'Infected', U for 'Uninfected'
11 for '11 dpi' and 33 for '33 dpi'

$$\begin{array}{cc} I & 11 \\ I & 11 \\ I & 11 \\ I & 33 \\ U & 33 \\ I & 33 \\ U & 11 \\ U & 11 \\ U & 11 \\ U & 33 \\ I & 33 \\ U & 33 \end{array} \quad \begin{pmatrix} \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \end{pmatrix} \quad \begin{pmatrix} \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \end{pmatrix}$$

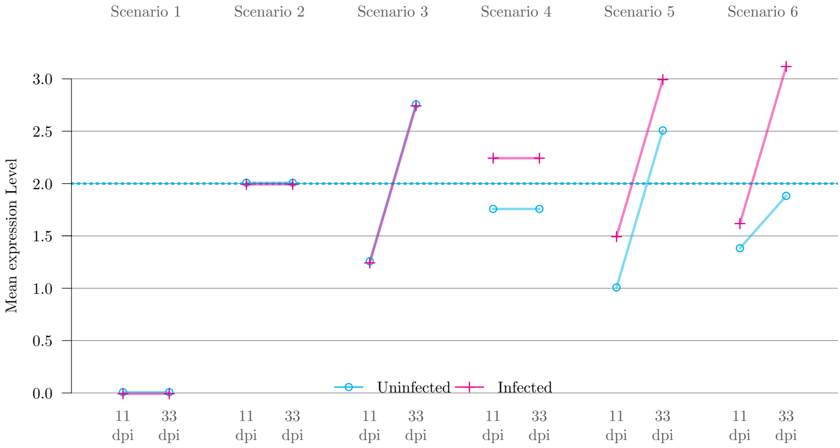
X matrix
(12 × 4)

Xβ vector
(p × 1)

Models with 2 factors: possible scenarios

2 factors:

- Status (2 levels): Uninfected and infected
- Time (2 levels): 11 and 33 dpi



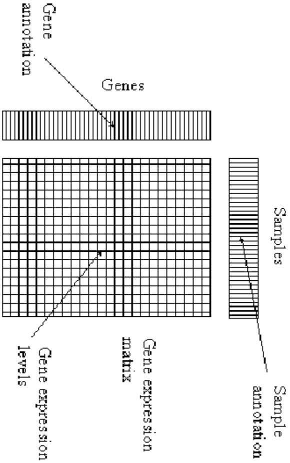
Design matrices for models with two two-level factors: R Code

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

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

Coffee break

Negative binomial regression: Model



$$\mathbf{y} \sim \mathsf{NB}(\boldsymbol{\mu}, \phi)$$

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

where

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

Negative binomial regression: Probability mass function

$$\mathbf{y} \sim \mathsf{NB}(\boldsymbol{\mu}, \phi)$$

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

with expectation and variance given by

- $\mathsf{E}[\mathbf{y}] = \boldsymbol{\mu} = \mathbf{s}\ 2^{\mathbf{X}\boldsymbol{\beta}}$
- $\mathsf{Var}[\mathbf{y}] = \boldsymbol{\mu}\left(1 + \frac{\boldsymbol{\mu}}{\phi}\right)$
- 2 parameters:
  - $\boldsymbol{\beta}$: regression coefficients
  - ϕ: shape/nuisance parameter

β_0-parameter: Interpretation of the intercept

```
> 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,name="Intercept")

log2 fold change (MLE): Intercept
Wald test p-value: Intercept
DataFrame with 1000 rows and 6 columns
      baseMean log2FoldChange     lfcSE      stat       pvalue         padj
     <numeric>      <numeric> <numeric> <numeric>    <numeric>    <numeric>
1      97.3140        6.90565  0.242562   28.4697 2.78073e-178 4.84448e-178
2     109.9860        6.89102  0.318468   21.6381 7.87448e-104 8.03519e-104
3      98.8111        6.57355  0.326862   20.1111  5.90379e-90  5.93346e-90
...        ...            ...       ...       ...          ...          ...
998    78.1356        6.57184  0.260146   25.2621 8.34043e-141 9.41358e-141
999    89.2920        6.05380  0.217898   27.7827 7.02445e-170 1.06593e-169
1000  103.5569        6.73029  0.246421   27.3122 3.03850e-164 4.29167e-164
```

- Mean expression level for gene ‘1’ for participants of condition ‘1’ (reference):

  $E[\mathbf{y}|\text{‘cond 1’}] = \widehat{\mu}_{\text{‘cond 1’}} = 2^{\widehat{\beta}_0} = 2^{6.90565} = 119.8969$
- $\widehat{\beta}_0 = \log_2(\widehat{\mu}_{\text{‘cond 1’}}) = \log_2(119.8969)$

β_1-parameter: Log2 fold change interpretation

```
> results(dds,name="cond_2_vs_1")

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
...        ...            ...       ...       ...       ...       ...
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
```

- $E[\mathbf{y}|\text{`cond 1}'] = \widehat{\mu}_{\text{`cond 1'}} = 2^{\widehat{\beta}_0}$
- $E[\mathbf{y}|\text{`cond 2}'] = \widehat{\mu}_{\text{`cond 2'}} = 2^{\widehat{\beta}_0+\widehat{\beta}_1} = 2^{\widehat{\beta}_0}2^{\widehat{\beta}_1}$

- If not DE → $\beta_1 = 0$ so that $\widehat{\mu}_{\text{`cond 2'}} = 2^{\widehat{\beta}_0}2^0 = 2^{\widehat{\beta}_0} = \widehat{\mu}_{\text{`cond 1'}}$,
- If DE → $\beta_1 \neq 0$ so that $\widehat{\mu}_{\text{`cond 2'}} = 2^{\widehat{\beta}_0}2^{\widehat{\beta}_1} = 2^{\widehat{\beta}_1}\widehat{\mu}_{\text{`cond 1'}}$.

Interpretation:

- $2^{\widehat{\beta}_1} = 2^{-0.682067} = 0.6232717$ is the *multiplicative/fold change in the mean expression level of participants of condition 2 compared to condition 1* so that $\widehat{\mu}_{\text{`cond 2'}} = 0.6232717 \times 119.8969 = 74.72831$
- $\widehat{\beta}_1$ is then the *log*$_2$ *fold change*.

β_1-parameter: Significance

```
> results(dds,name="cond_2_vs_1")

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
...        ...            ...       ...       ...       ...       ...
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
```

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

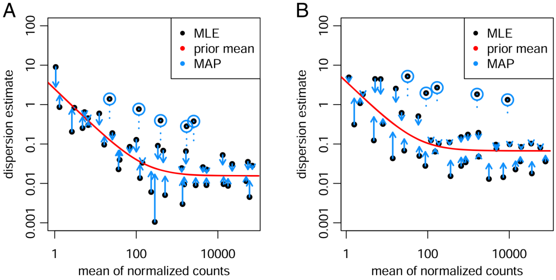
- **H0:** $\beta_1 = 0$ versus **H1:** $\beta_1 \neq 0$
- Z-statistic $= \frac{\widehat{\beta}_1}{\widehat{\sigma}_{\widehat{\beta}_1}} = \frac{-0.682067}{0.344525} = -1.979730$
- P-value with $Z \sim N(0,1)$ under **H0** is given by

```
> 2*(1-pnorm(abs(-1.979730)))

[1] 0.04773388
```

ϕ-parameter: 3 Estimators

- gene-wise shape/dispersion parameter estimates (black dots)
  not efficient
- assuming a smooth non-linear fit between mean and shape (red line)
  strong assumption: borrow information from neighbouring genes assuming a similar mean/shape relationship,
- Bayesian combination of both [mid-way optimal solution].



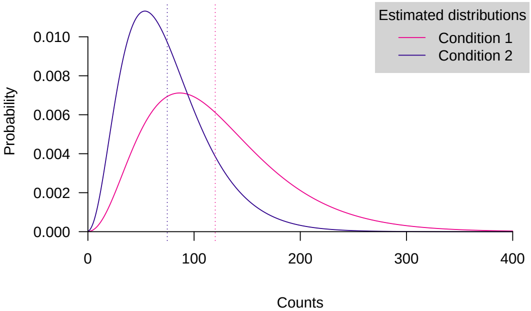
(Love et al (2015))

Negative binomial regression: Assumed Distribution

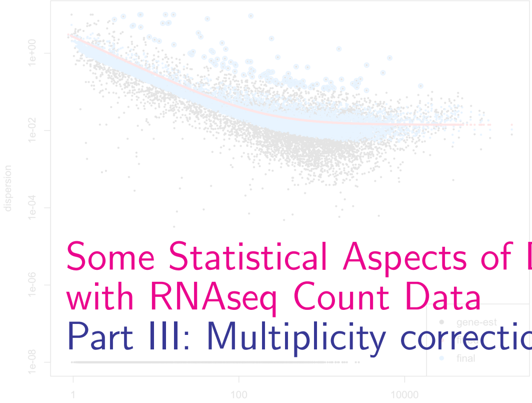
```
-> mcols(dds)[,c("Intercept","cond_2_vs_1","dispGeneEst","dispFit","dispersion")]

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
  $\mathbf{y} \sim \text{NB}(\widehat{\mu}_{\text{`cond 1'}} = 2^{6.90565} = 119.8969, \widehat{\phi} = 0.274708)$
- For gene 1 and condition 2, we have
  $\mathbf{y} \sim \text{NB}(\widehat{\mu}_{\text{`cond 2'}} = 2^{6.90565}2^{-0.682067} = 74.72831, \widehat{\phi} = 0.274708)$



Coffee break



Some Statistical Aspects of DE Analysis with RNAseq Count Data
Part III: Multiplicity correction

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

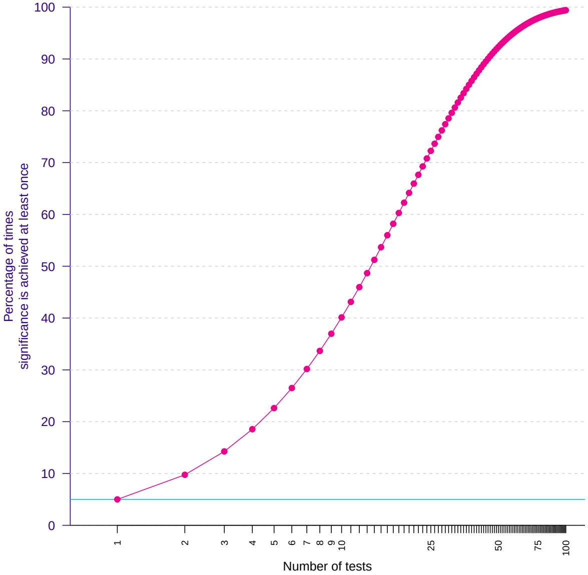
(Source: G. Marot, INRIA)

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

one dispersion per gene

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

Multiplicity correction: Familywise error rate



Multiplicity correction

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.

The False Discovery Rate (FDR)

Idea : Do not control the error rate but the proportion of error
$\Rightarrow$ 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)
```

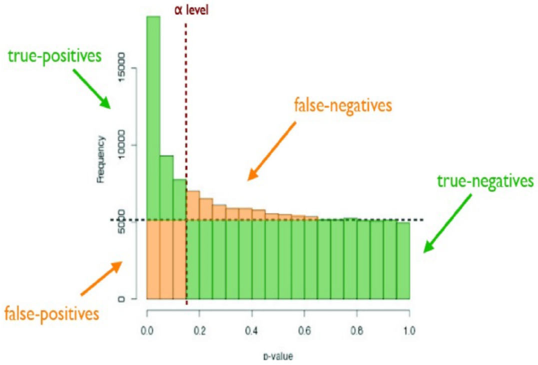
```
log2 fold change (MLE): cond 2 vs 1
Wald test p-value: cond 2 vs 1
DataFrame with 1000 rows and 6 columns
      baseMean log2FoldChange     lfcSE      stat    pvalue      padj
     <numeric>      <numeric> <numeric> <numeric> <numeric> <numeric>
1      97.3140      -0.682067  0.344525 -1.979730 0.0477339  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

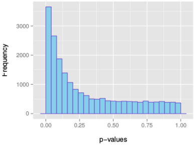
Standard assumption for p-value distribution

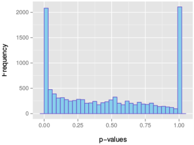


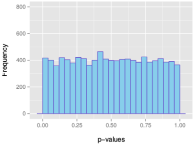
Source : M. Guedj, Pharnext

p-values histograms for diagnosis

Examples of expected overall distribution







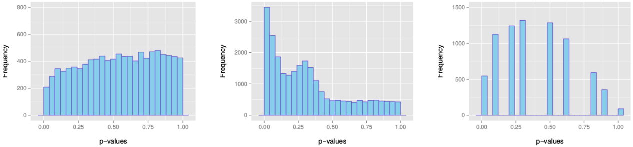
(a) : the most desirable shape

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

(c) : do not expect positive tests after correction

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
```

Adjusted p-values valid if
1/ counts of each gene follow an homomorphic Gamma mixture of Poisson distribution (Negative binomial) per condition with mean to dispersion relationship similar to the one of neighbouring genes,
2/ the sample size if large enough for the asymptotic theory to hold for Wald Z-tests,
3/ assumptions of the chosen multiplicity correction hold (PRDSH0)