Pseudo-replication: Mixed Models

Lieven Clement

Note, that this dataset has been acquired with DDA (data dependent acquisition). It was searched with MaxQuant. The analysis starts from the peptides.txt file, which is a table in the output directory of MaxQuant with the data summarized at the peptide-level.

The preprocessing steps are slightly different. The modeling concepts, however, are also applicable to DIA data.

Background

A study on the facultative pathogen Francisella tularensis was conceived by Ramond et al. (2015). F. tularensis enters the cells of its host by phagocytosis. The authors showed that F. tularensis is arginine deficient and imports arginine from the host cell via an arginine transporter, ArgP, in order to efficiently escape from the phagosome and reach the cytosolic compartment, where it can actively multiply. In their study, they compared the proteome of wild type F. tularensis (WT) to ArgP-gene deleted F. tularensis (knock-out, D8).

For this exercise, we use the F. tularensis dataset where bacterial cultures were grown in biological triplicate for each genotype and each biorep was run in technical triplicate on a nanoRSLC-Q Exactive PLUS instrument. The data were searched with MaxQuant version 1.4.1.2. and are available on the PRIDE repository: PXD001584.

- 2 genotypes (WT, D8) $\times$ 3 biorepeats $\times$ 3 technical repeats (MS-runs) = 18 runs
- The technical repeats are **pseudo-replicates**: they are not independent samples!

Import Data and Preprocessing

Data

Click to see code

1. Load libraries

```
library(tidyverse)
library(limma)
library(QFeatures)
library(msqrob2)
library(plotly)
library(ExploreModelMatrix)
```

2. We use a peptides.txt file from MS-data quantified with maxquant that contains MS1 intensities summarized at the peptide level, and an annotation file with the design. Both are available in the msqrob2data GitHub repository.

```
library("BiocFileCache")
bfc <- BiocFileCache()
peptidesFile <- bfcrpath(bfc, "https://github.com/statOmics/msqrob2data/raw/refs/heads/main/
annotationFile <- bfcrpath(bfc, "https://github.com/statOmics/msqrob2data/raw/refs/heads/mai
```

3. The annotation file contains the column `quantCols` with the names of the columns in the peptides file that hold the intensity data for the different samples, and the design variables `genotype` and `biorep`.

```
annot <- read.csv2(annotationFile)
annot$genotype <- annot$genotype |>
  as.factor() |>
  relevel("WT")
annot$biorep <- as.factor(annot$biorep)
annot |>
  knitr::kable()
```

quantCols	genotype	biorep	sampleId
Intensity.1WT_20_2h_n3_1	WT	wt_n3	wt_n3_1
Intensity.1WT_20_2h_n3_2	WT	wt_n3	wt_n3_2
Intensity.1WT_20_2h_n3_3	WT	wt_n3	wt_n3_3
Intensity.1WT_20_2h_n4_1	WT	wt_n4	wt_n4_1
Intensity.1WT_20_2h_n4_2	WT	wt_n4	wt_n4_2
Intensity.1WT_20_2h_n4_3	WT	wt_n4	wt_n4_3
Intensity.1WT_20_2h_n5_1	WT	wt_n5	wt_n5_1
Intensity.1WT_20_2h_n5_2	WT	wt_n5	wt_n5_2
Intensity.1WT_20_2h_n5_3	WT	wt_n5	wt_n5_3
Intensity.3D8_20_2h_n3_1	D8	d8_n3	d8_n3_1
Intensity.3D8_20_2h_n3_2	D8	d8_n3	d8_n3_2
Intensity.3D8_20_2h_n3_3	D8	d8_n3	d8_n3_3

quantCols	genotype	biorep	sampleId
Intensity.3D8_20_2h_n4_1	D8	d8_n4	d8_n4_1
Intensity.3D8_20_2h_n4_2	D8	d8_n4	d8_n4_2
Intensity.3D8_20_2h_n4_3	D8	d8_n4	d8_n4_3
Intensity.3D8_20_2h_n5_1	D8	d8_n5	d8_n5_1
Intensity.3D8_20_2h_n5_2	D8	d8_n5	d8_n5_2
Intensity.3D8_20_2h_n5_3	D8	d8_n5	d8_n5_3

4. Read the data and store it in QFeatures object

```
qf <- readQFeatures(
  assayData = read.delim(peptidesFile),
  colData = annot,
  fnames = 1,
  name = "peptideRaw")
```

Preprocessing

Click to see code to preprocess the data

1. Log transform

- peptides with zero intensities are missing peptides and should be represented with a `NA` value rather than `0`.

```
qf <- zeroIsNA(qf, "peptideRaw") # convert 0 to NA
```

- Logtransform data with base 2

```
qf <- logTransform(qf, base = 2, i = "peptideRaw", name = "peptides_log")
```

2. Filtering

- Only use proteotypic proteins

```
qf <- filterFeatures(qf, ~ !grepl(pattern = ";", Proteins))
```

- Remove reverse sequences (decoys) and contaminants.

```
qf <- filterFeatures(qf, ~ Reverse != "+")
qf <- filterFeatures(qf, ~ Contaminant != "+")
```

- Drop peptides that were only identified in a single biorepeat. Note, that in experiments without technical repeats we filter on the number of samples in which a peptide is picked up. Here, every biorep is assessed in technical triplicate, so we require that a peptide is picked up in at least two biorepeats.

```
rowData(qf[["peptides_log"]])$nObsBiorep <- apply(
  assay(qf[["peptides_log"]]),
  1,
  function(intensity)
    colData(qf)$biorep[!is.na(intensity)] |>
    unique() |>
    length())

qf <- filterFeatures(qf, ~ nObsBiorep >= 2, keep = TRUE)
```

We keep 7542 peptides upon filtering.

3. Normalization using median of ratios

```
qf <- sweep( #4. Subtract log2 norm factor column-by-column (MARGIN = 2)
  qf,
  MARGIN = 2,
  STATS = nfLogMedianOfRatios(qf, i = "peptides_log"), #1.
  i = "peptides_log",
  name = "peptides_norm"
)
```

4. Summarization. We use maxLFQ from the iq package.

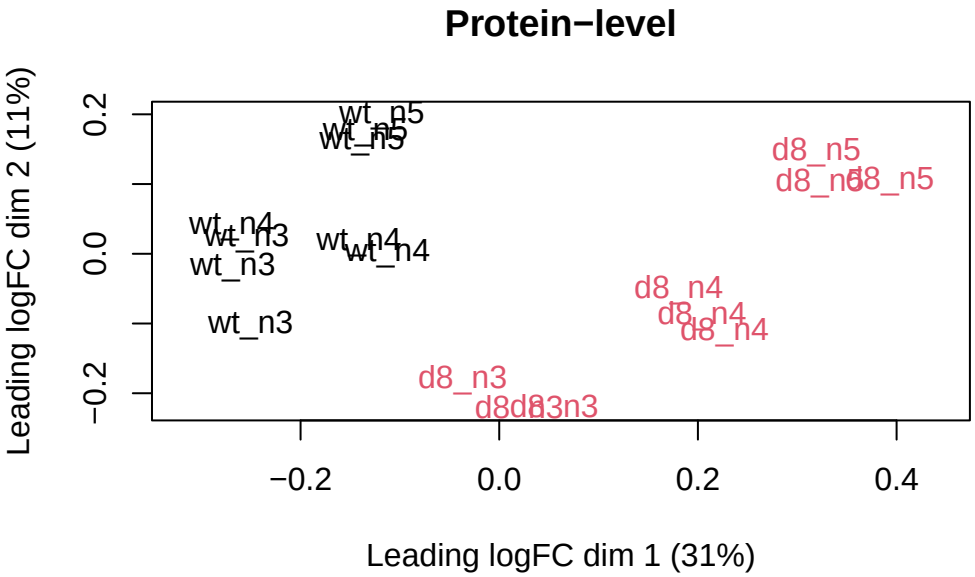
```
qf <- aggregateFeatures(qf,
    i = "peptides_norm",
    fcol = "Proteins",
    name = "proteins",
    fun = function(X) iq::maxLFQ(X)$estimate
)
```

```
  |
  |                                                                      |   0%
  |
  |======================================================================| 100%
```

Data exploration

We can visualize our data using a Multi Dimensional Scaling plot, eg. as provided by the `limma` package.

```
qf[["proteins"]] |>
  assay() |>
  limma::plotMDS(
    col = as.numeric(colData(qf)$genotype),
    labels = colData(qf)$biorep,
    main = "Protein-level")
```



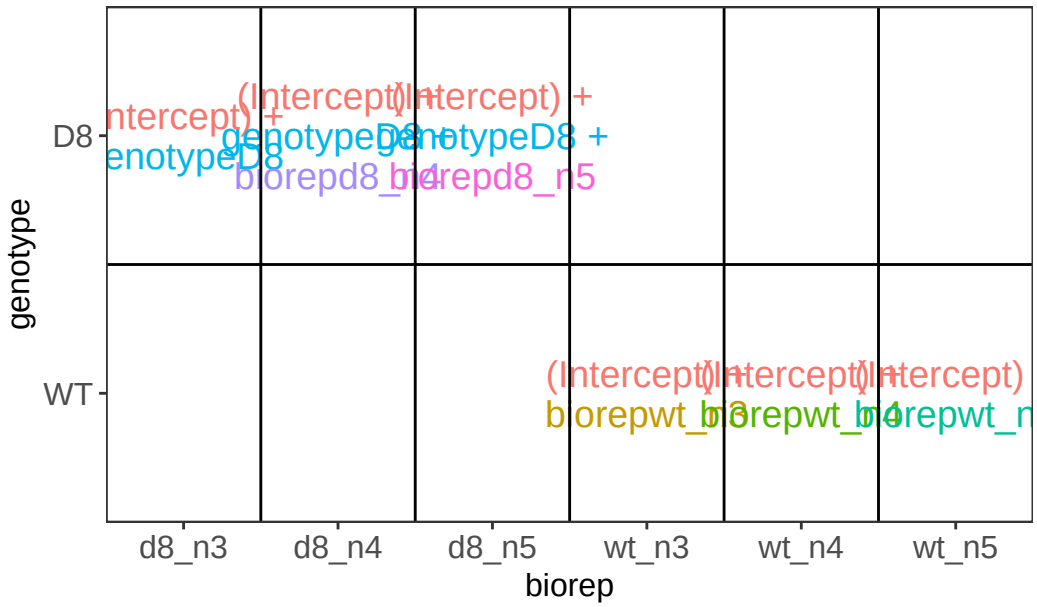
- The first axis in the plot is showing the leading log fold changes (differences on the log scale) between the samples.
- The samples show a clear separation according to the genotype in the first dimension of the MDS plot.
- The samples of the three MS-runs from the same biorepeat cluster together, indicating that biorepeat is a leading source of variability.

Data Analysis

Design

```
VisualizeDesign(colData(qf), ~ genotype + biorep)$plotlist
```

```
[[1]]
```



Unlike the mouse example in the blocking wrap-up, we cannot correct for biorepeat using a randomized complete block analysis. Indeed, once we condition on biorepeat, we also condition on the genotype: biorep is nested within genotype. Hence, when inferring on the treatment effect we should account for the variability both within and between biorepeats.

> Think about it: fish in tanks
>
> Researchers want to assess the effect of three diets on the weight gain of fish. They have 12 tanks with the same number of fish. Each diet is randomly assigned to 4 tanks, and at the end of the experiment the weight gain of 6 fish in each tank is recorded, i.e. 72 weight gains.
>
> - On which unit is the diet randomised (the experimental unit)? On which unit is

the weight gain measured (the observational unit)?
- How many independent replicates are there per diet?
- Can the fish of the same tank be considered as independent repeats? Why not?
- What corresponds to the diet, the tanks and the fish in the Francisella experiment?

Click to see the answer

- The diet is randomised to the tanks: the tanks are the experimental units. The weight gain is measured on the fish: the fish are the observational units.
- There are only 4 independent replicates per diet, not 24.
- No, fish in the same tank share the same environment (water quality, temperature, feeding, …) and their weight gains are correlated. The fish of the same tank are pseudo-replicates.
- In the Francisella experiment the genotype plays the role of the diet, the bio-repeats (bacterial cultures) are the tanks and the MS-runs are the fish:

Fish experiment	Francisella experiment
diet	genotype
tank: experimental unit	bio-repeat: experimental unit
fish: observational unit	MS-run: observational unit
4 tanks per diet	3 bio-repeats per genotype
6 fish per tank	3 MS-runs per bio-repeat

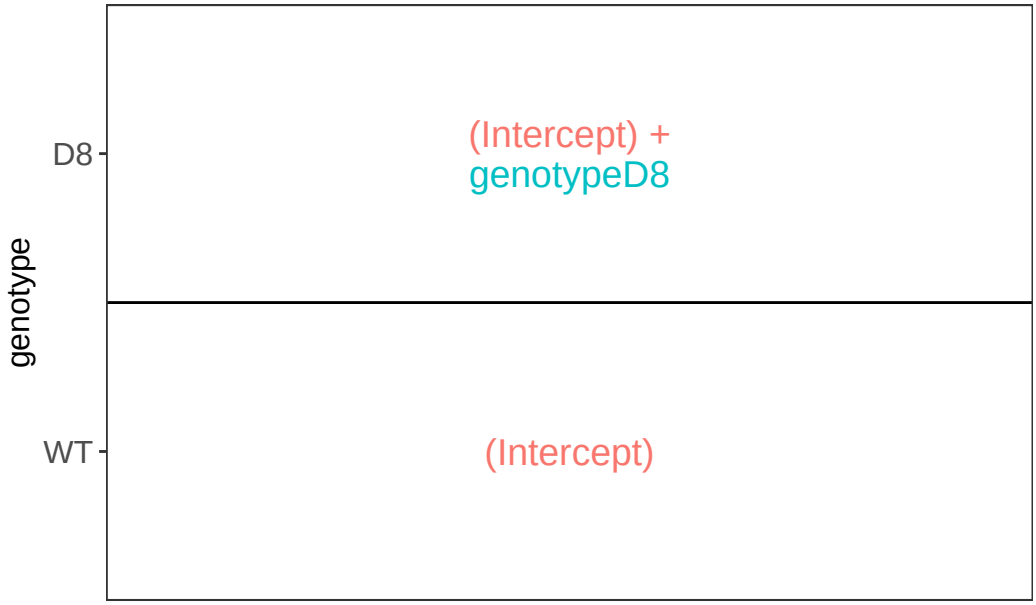
Wrong analysis

In the original paper, the authors simply ignored the correlation in the data according to biorepeat and INCORRECTLY acted as if each technical repeat was an independent sample. So they act as if they have a 9 versus 9 group comparison instead of correctly accounting that they only have three independent biorepeats for each genotype.

We will fit this WRONG model to see how this impacts the data analysis.

```
VisualizeDesign(colData(qf), ~ genotype)$plotlist
```

```
[[1]]
```



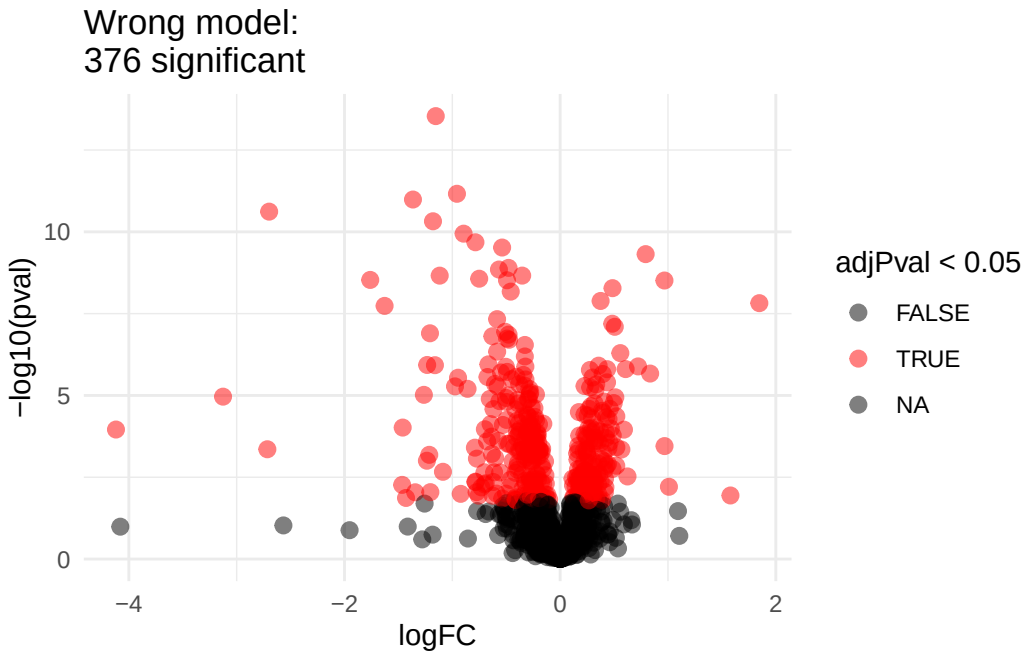

```
qf <- msqrob(
  object = qf,
  i = "proteins",
  formula = ~ genotype,
  modelColumnName = "wrongModel")
```

Statistical inference: log2FC between D8 and WT is quantified by `genotypeD8`.

```
L <- makeContrast("genotypeD8 = 0", parameterNames = c("genotypeD8"))
qf <- hypothesisTest(
  object = qf,
  i = "proteins",
  contrast = L,
  modelColumn = "wrongModel",
  resultsColumnNamePrefix = "wrongModel-")
```

```
volcanoWrong <- ggplot(
    rowData(qf[["proteins"]])$`wrongModel-genotypeD8`,
    aes(x = logFC, y = -log10(pval), color = adjPval < 0.05)
) +
    geom_point(cex = 2.5) +
    scale_color_manual(values = alpha(c("black", "red"), 0.5)) +
```

```
    theme_minimal() +
    ggtitle(paste0("Wrong model: \n",
                   sum(rowData(qf[["proteins"]])$`wrongModel-genotypeD8`$adjPval < 0.05, na.rm =
            " significant"))
volcanoWrong
```



Note, that 376 proteins are found to be differentially abundant according to the WRONG DATA ANALYSIS that ignores the fact that intensities from technical repeats (MS-runs) from the same biorepeat are correlated.

As the analysis acts as if there are 9 independent repeats for each genotype, we can expect the analysis to be overly liberal. So we can expect many false positive results.

Correct data analysis

We model the protein level expression values using `msqrob`. By default `msqrob2` estimates the model parameters using robust regression.

We will model the data with a different group mean. The group is encoded in the variable `genotype` of the colData. We will also have to include a random effect for bio-repeat to address the pseudo-replication in the experiment. Indeed, the data from the same bio-repeat will be correlated!

We can specify this model by using a formula with the factor `genotype` as a fixed effect and the factor `biorep` as a random effect: `formula = ~ genotype + (1|biorep)`.

```
qf <- msqrob(
  object = qf,
  i = "proteins",
  formula = ~ genotype + (1|biorep))
```

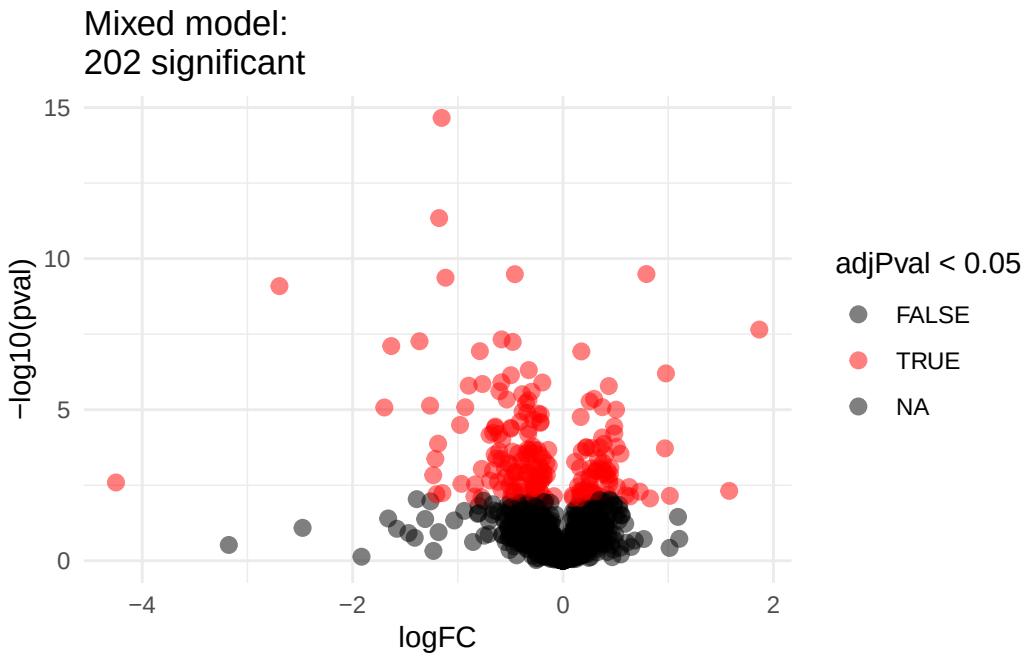
For this model we can also use the same contrast. Indeed, the model now contains a parameter for the intercept, genotype and a random intercept for each biorepeat.

```
getCoef(rowData(qf[["proteins"]])$msqrobModels[[1]])
```

```
           (Intercept)             genotypeD8 (Intercept)biorepd8_n3
           26.16888331            -0.06186638            -0.16725774
(Intercept)biorepd8_n4 (Intercept)biorepd8_n5 (Intercept)biorepwt_n3
            0.05761904             0.10963871            -0.12772012
(Intercept)biorepwt_n4 (Intercept)biorepwt_n5
           -0.01520489             0.14292502
```

```
qf <- hypothesisTest(object = qf, i = "proteins", contrast = L)
```

```
volcanoMixed <- ggplot(
    rowData(qf[["proteins"]])$genotypeD8,
    aes(x = logFC, y = -log10(pval), color = adjPval < 0.05)
) +
    geom_point(cex = 2.5) +
    scale_color_manual(values = alpha(c("black", "red"), 0.5)) +
    theme_minimal() +
    ggtitle(paste0("Mixed model: \n",
                sum(rowData(qf[["proteins"]])$genotypeD8$adjPval < 0.05, na.rm = TRUE),
            " significant"))
volcanoMixed
```



Note, that 202 proteins are found to be differentially abundant. Hence, when we address the correlation structure in the data, the analysis returns far fewer DA proteins than the incorrect analysis of the original authors, who acted as if all technical replicates were independent repeats.

Mixed model

The mixed model addresses the correlation structure in the data by including random biorepeat effects:

$$\begin{cases} y_i &= \beta_0 + \beta_{D8} x_{D8,i} + u_1 z_{b1,i} + u_2 z_{b2,i} + u_3 z_{b3,i} + u_4 z_{b4,i} + u_5 z_{b5,i} + u_6 z_{b6,i} + \epsilon_i \\ u_j &\sim N(0, \sigma_u^2) \\ \epsilon_i &\sim N(0, \sigma_\epsilon^2) \end{cases},$$

with u_j random effects that model the correlation in the intensities from technical repeats (MS-runs) that originate from the same bio-repeat. These random intercepts are assumed to be i.i.d. normally distributed with mean 0 and variance σ_u^2. Hence, σ_u^2 models the between biorepeat variability and σ_ϵ^2 the within biorepeat variability.

We can also write the model in matrix form

$$\mathbf{Y} = \mathbf{X}\beta + \mathbf{Z}\mathbf{u} + \epsilon$$

with

$$\mathbf{Y} = \begin{bmatrix} y_1 \\ \vdots \\ y_{18} \end{bmatrix}, \mathbf{X} = \begin{bmatrix} 1 & x_{D8,1} \\ \vdots & \vdots \\ 1 & x_{D8,18} \end{bmatrix}, \beta = \begin{bmatrix} \beta_0 \\ \beta_{D8} \end{bmatrix}, \mathbf{Z} = \begin{bmatrix} z_{b1,1} & \cdots & z_{b6,1} \\ \vdots & & \vdots \\ z_{b1,18} & \cdots & z_{b6,18} \end{bmatrix}, \mathbf{u} = \begin{bmatrix} u_1 \\ \vdots \\ u_6 \end{bmatrix}$$

Hence, with the mixed model, the variance covariance matrix of the intensities becomes

$$\begin{aligned} \Sigma_{\mathbf{Y}} &= \text{var}[\mathbf{Y}] \\ &= \text{var}[\mathbf{X}\beta + \mathbf{Z}\mathbf{u} + \epsilon] \\ &= \mathbf{Z}\text{var}[\mathbf{u}]\mathbf{Z}^T + \mathbf{I}\sigma_\epsilon^2 \\ &= \mathbf{Z}\mathbf{Z}^T\sigma_u^2 + \mathbf{I}\sigma_\epsilon^2 \end{aligned}$$

```
X <- model.matrix(~ genotype, colData(qf))
Z <- model.matrix(~ -1 + biorep, colData(qf))
rownames(Z) <- colData(qf)$biorep
Z %*% t(Z)
```

	wt_n3	wt_n3	wt_n3	wt_n4	wt_n4	wt_n4	wt_n5	wt_n5	wt_n5	d8_n3	d8_n3	d8_n3
wt_n3	1	1	1	0	0	0	0	0	0	0	0	0
wt_n3	1	1	1	0	0	0	0	0	0	0	0	0
wt_n3	1	1	1	0	0	0	0	0	0	0	0	0
wt_n4	0	0	0	1	1	1	0	0	0	0	0	0
wt_n4	0	0	0	1	1	1	0	0	0	0	0	0
wt_n4	0	0	0	1	1	1	0	0	0	0	0	0
wt_n5	0	0	0	0	0	0	1	1	1	0	0	0
wt_n5	0	0	0	0	0	0	1	1	1	0	0	0
wt_n5	0	0	0	0	0	0	1	1	1	0	0	0
d8_n3	0	0	0	0	0	0	0	0	0	1	1	1
d8_n3	0	0	0	0	0	0	0	0	0	1	1	1
d8_n3	0	0	0	0	0	0	0	0	0	1	1	1
d8_n4	0	0	0	0	0	0	0	0	0	0	0	0
d8_n4	0	0	0	0	0	0	0	0	0	0	0	0
d8_n4	0	0	0	0	0	0	0	0	0	0	0	0
d8_n5	0	0	0	0	0	0	0	0	0	0	0	0
d8_n5	0	0	0	0	0	0	0	0	0	0	0	0
d8_n5	0	0	0	0	0	0	0	0	0	0	0	0

	d8_n4	d8_n4	d8_n4	d8_n5	d8_n5	d8_n5
wt_n3	0	0	0	0	0	0
wt_n3	0	0	0	0	0	0
wt_n3	0	0	0	0	0	0

wt_n4	0	0	0	0	0	0
wt_n4	0	0	0	0	0	0
wt_n4	0	0	0	0	0	0
wt_n5	0	0	0	0	0	0
wt_n5	0	0	0	0	0	0
wt_n5	0	0	0	0	0	0
d8_n3	0	0	0	0	0	0
d8_n3	0	0	0	0	0	0
d8_n3	0	0	0	0	0	0
d8_n4	1	1	1	0	0	0
d8_n4	1	1	1	0	0	0
d8_n4	1	1	1	0	0	0
d8_n5	0	0	0	1	1	1
d8_n5	0	0	0	1	1	1
d8_n5	0	0	0	1	1	1

So, we see that the correlation of the data from the same biological repeat is correctly addressed and that the data from distinct biorepeats are assumed to be independent. Hence, the variance-covariance matrix of $\mathbf{Y}$ has a block diagonal structure, with as variance $\sigma_u^2 + \sigma_\epsilon^2$ and the covariance between intensities from the same biorepeat equals σ_u^2.

$$\Sigma_{\mathbf{Y}} = \begin{bmatrix} \sigma_u^2+\sigma_\epsilon^2 & \sigma_u^2 & \sigma_u^2 & 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ \sigma_u^2 & \sigma_u^2+\sigma_\epsilon^2 & \sigma_u^2 & 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ \sigma_u^2 & \sigma_u^2 & \sigma_u^2+\sigma_\epsilon^2 & 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_u^2+\sigma_\epsilon^2 & \sigma_u^2 & \sigma_u^2 & \dots & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_u^2 & \sigma_u^2+\sigma_\epsilon^2 & \sigma_u^2 & \dots & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_u^2 & \sigma_u^2 & \sigma_u^2+\sigma_\epsilon^2 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & \dots & \sigma_u^2+\sigma_\epsilon^2 & \sigma_u^2 & \sigma_u^2 \\ 0 & 0 & 0 & 0 & 0 & 0 & \dots & \sigma_u^2 & \sigma_u^2+\sigma_\epsilon^2 & \sigma_u^2 \\ 0 & 0 & 0 & 0 & 0 & 0 & \dots & \sigma_u^2 & \sigma_u^2 & \sigma_u^2+\sigma_\epsilon^2 \end{bmatrix}$$

Ramond, Elodie, Gael Gesbert, Ida Chiara Guerrera, et al. 2015. "Importance of Host Cell Arginine Uptake in Francisella Phagosomal Escape and Ribosomal Protein Amounts." *Mol. Cell. Proteomics* 14 (4): 870–81.