Practical 5: Mixed models

BIOM4025 - Statistical Modelling

Erik Postma - e.postma@exeter.ac.uk

Weeks 9-10: 18 - 29 November 2024

Contents

1 Introduction **1**

2 Getting started **2**
2.1 Importing the data 2
2.2 Some descriptive statistics 2

3 Is PVL sperm different? **5**
3.1 t-test 5
3.2 Linear model 7
3.3 Linear mixed model 8

4 Is PVL sperm less variable? **12**
4.1 Calculating and comparing coefficients of variation 12
4.2 A mixed model approach 14

1 Introduction

In internal fertilisers, less than 1% of the inseminated sperm actually reach the site of fertilisation. This reduction in sperm numbers is assumed to result from oviductal selection, ensuring only the 'fittest' sperm are able to fertilise the egg. But what makes for 'fit' sperm? Are the sperm that make it to the site of fertilisation in any way special?

In this study, Hemmings *et al.* tackle this question in a captive population of zebra finches (*Taeniopygia guttata*). In birds, the sperm that reach the ovum after it has been fertilised are trapped by the outer perivitelline layer (PVL). This makes it possible to measure the length of the head, midpiece and tail of PVL sperm, which we know has reached the ovum, and compare this to a random sample of sperm obtained from the male's faeces.

> **HOMEWORK**
> To familiarise yourself with the topic and study system, read the paper that accompanies the data set we will be using for this practical: Hemmings *et al.*, 2016. Intra-ejaculate sperm selection in female zebra finches. *Biology Letters* **12**: 20160220. http://dx.doi.org/10.1098/rsbl.2016.0220. A PDF of the paper can also be found on the ELE page.

Many journals, including *Biology Letters*, have made it compulsory to make the data on which an article is based publicly available. This is good news for us, as it allows us to reanalyse their data and compare our results to those presented in the paper. For this article, the data can be downloaded from the Dryad Digital Repository or from the ELE page.

NOTE
Only have a look at the **HARD EXERCISES** (and especially the **VERY HARD EXERCISES**!) once you have finished with all the others, and don't worry if you can't solve do them: They are pretty hard!

2 Getting started

2.1 Importing the data

Copy the data file into a folder on your computer and have a look at it using *Excel* or the text editor of your choice. Do the column names contain any spaces? Are they short but informative? Do any of the columns contain missing values? Make any changes you think are necessary. Note that below I may use slightly different column names from you, so be careful when copy-pasting any of the code.

Once you are happy with what your data file looks like, import it into `R`. You will have to tell `R` where to find the file using `setwd()` and then use `read.csv()` (a version of `read.table()` that is more convenient when importing comma-separated text files) to read it into `R`:

```
setwd("/This/Is/Path/To/My/Working/Directory")
s.data <- read.csv("faecal and pvl sperm morphology data.csv", stringsAsFactors=TRUE)
```

Have the data imported correctly? Are all the rows and columns there? Are the columns containing numbers `numeric` and the others of type `character` or `factor`? Any other data types?

2.2 Some descriptive statistics

Let's first check whether the sample sizes reported in the paper match those in the data file.

The number of sperm that were measured:

```
nrow(s.data)
```

```
[1] 813
```

The number of males included in the analysis:

```
length(unique(s.data$bird.no))
```

```
[1] 27
```

The number of sperm analysed per `bird.no` and per `sample.type`:

```
table <- table(s.data$sample.type, s.data$bird.no)
table
```

```
         1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
  faecal 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10
  pvl    16 33 11 18 10 12 20 43 11 18 10 23 19 12 29 14 35 17 33 16 18 28 14

         24 25 26 27
  faecal 10 10 10 10
  pvl    12 20 36 15
```

```
summary(table[1, ]) # Number of faecal sperm
```

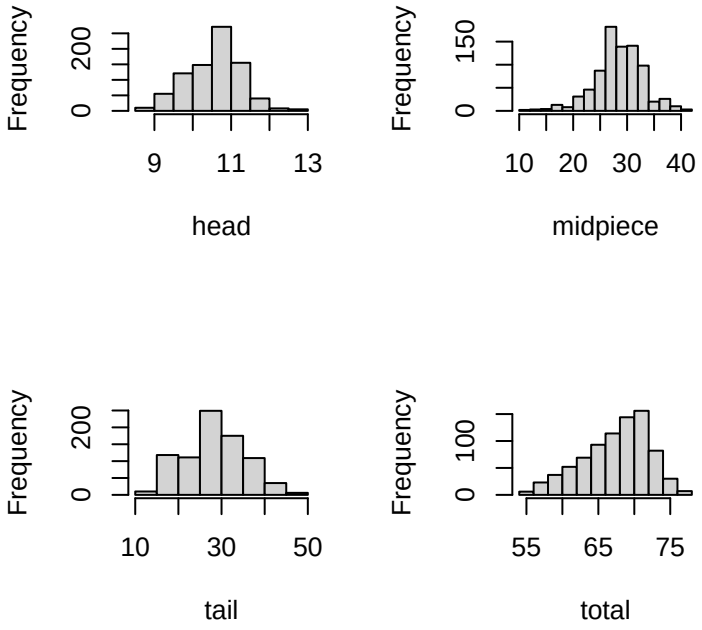
```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
     10      10      10      10      10      10
```

```
summary(table[2, ]) # Number of pvl sperm
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  10.00   13.00   18.00   20.11   25.50   43.00
```

Also, it is probably a good idea to have look at the distribution of the different sperm traits:

```
par(mfrow=c(2,2))
hist(s.data$head.length, xlab="head", main=NULL)
hist(s.data$midpiece.length, xlab="midpiece", main =NULL)
hist(s.data$tail.length, xlab="tail", main=NULL)
hist(s.data$total.length, xlab="total", main =NULL)
```



EXERCISE 5.1
Looking at these distributions, what do you conclude with respect to their distribution?

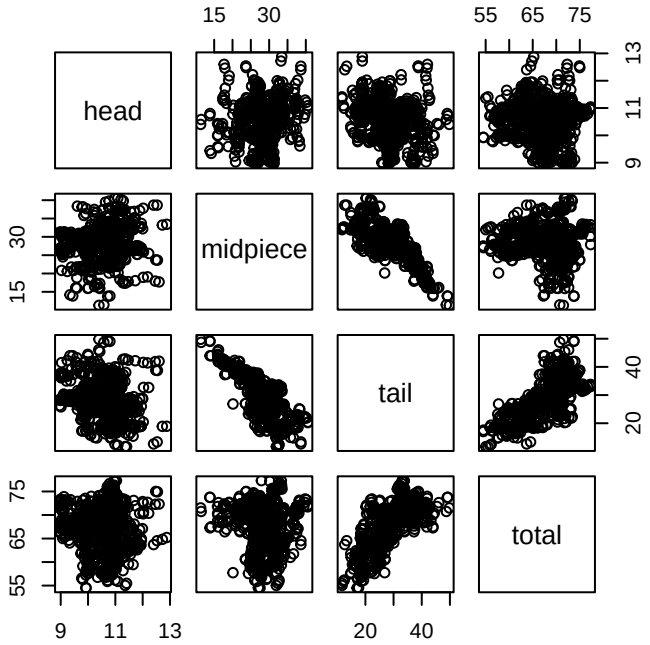
Finally, with four different traits, we should check whether they are correlated. First, we can calculate all pairwise correlation coefficients for the four sperm traits:

```
cor(s.data[, 3:6])
```

```
                head.length midpiece.length tail.length total.length
head.length      1.00000000       0.1710997  -0.2389568  -0.03615098
midpiece.length  0.17109971       1.0000000  -0.7335127  -0.10645707
tail.length     -0.23895681      -0.7335127   1.0000000   0.74174389
total.length    -0.03615098      -0.1064571   0.7417439   1.00000000
```

Although we could create all pairwise plots one-by-one, the `pairs()` function provides a convenient way of plotting all of them at once:

```
pairs(cbind(s.data$head.length, s.data$midpiece.length, s.data$tail.length,
            s.data$total.length),
            labels = c("head", "midpiece", "tail", "total"))
```



Finally, we could test if any of these correlations are statistically significant using `cor.test()`. For example:

```
cor.test(s.data$head.length, s.data$midpiece.length)
```

```
    Pearson's product-moment correlation

data:  s.data$head.length and s.data$midpiece.length
t = 4.9455, df = 811, p-value = 9.233e-07
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.1035606 0.2370682
sample estimates:
      cor
0.1710997
```

```
cor.test(s.data$tail.length, s.data$total.length)
```

```
    Pearson's product-moment correlation

data:  s.data$tail.length and s.data$total.length
t = 31.495, df = 811, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.7091536 0.7711713
sample estimates:
      cor
0.7417439
```

EXERCISE 5.2
Which traits show the strongest correlations? Does this come as a surpise to you?

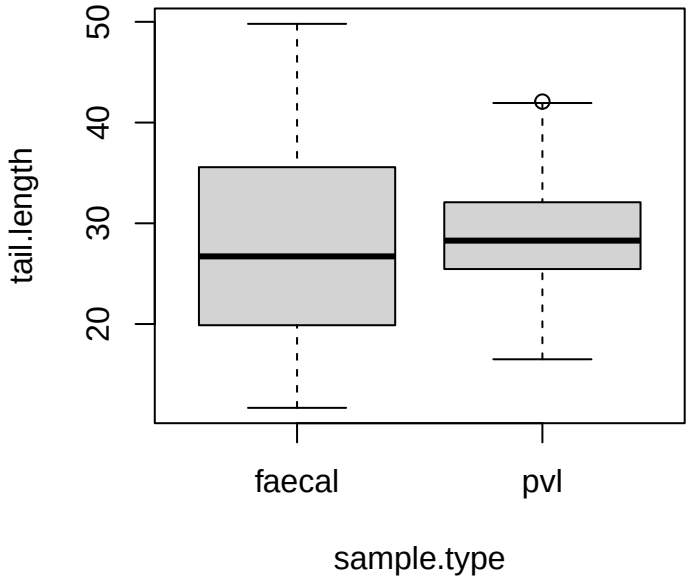
EXERCISE 5.3
Calculate the total length yourself, and compare this to `total.length`. Do the two measures of total length match?

3 Is PVL sperm different?

We would like to know if, on average, PVL sperm is different from the sperm collected from the faeces, which is assumed to be a random sample of all sperm.

In a first step, let's plot `tail.length` against `sample.type`. Because `sample.type` is a factor (we used `stringsAsFactors=TRUE` when importing the data), this provides us with a box plot:

```
par(mfrow=c(1,1))
plot(tail.length ~ sample.type, data=s.data)
```



Judging from this plot, PVL sperm tends to have a slightly longer tail. However, there appears to be a lot of variation within each group, so is this difference statistically significant?

3.1 t-test

To test this, we could perform a simple t-test. Note that by default, `t.test()` performs a Welch two sample t-test, which does not make the assumption that the variances in both groups are equal. Based on the box plot, and the fact that we expect the variance to be lower in PVL sperm (see Section 4), this seems like a good idea.

```
t.test(tail.length ~ sample.type, data = s.data)
```

```
    Welch Two Sample t-test

data:  tail.length by sample.type
t = -0.92994, df = 379.63, p-value = 0.353
alternative hypothesis: true difference in means between group faecal and group pvl is not equal to 0
95 percent confidence interval:
 -1.7020024  0.6090022
sample estimates:
mean in group faecal    mean in group pvl
            28.08015             28.62665
```

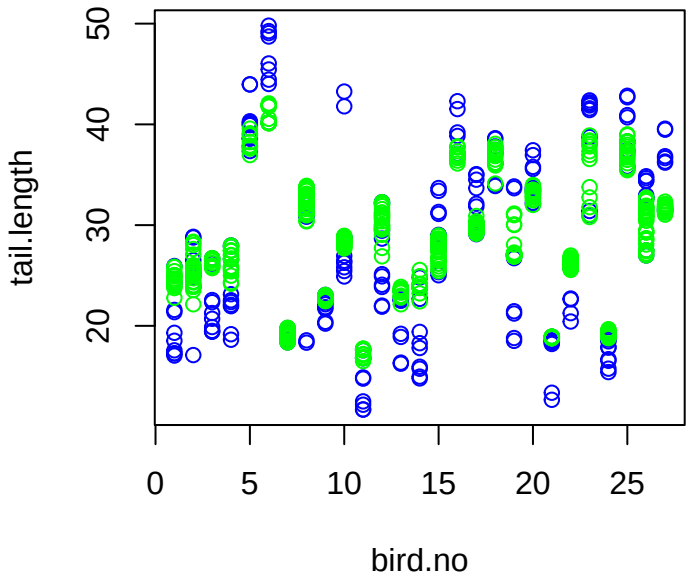
Based on this test, the difference in length is far from statistically significant.

EXERCISE 5.4
Use `lm()` to test for a difference in `tail.length` between `pvl` and `faecal` sperm. How do they

compare to the output provided by `t.test()`?

Before we conclude that there is indeed no difference in `tail.length` between `pvl` and `faecal` sperm, let's have a closer look at the data by plotting the individual measurements for both sample types per male:

```
plot(tail.length ~ bird.no, data=s.data, pch=NA)
points(tail.length ~ bird.no, data=s.data[s.data$sample.type=="faecal", ], col="blue")
points(tail.length ~ bird.no, data=s.data[s.data$sample.type=="pvl", ], col="green")
```



Looking at this plot it is clear that there is a lot more variation in `tail.length` among males then there is within the ejaculate of a single male, and this between-male variation may be masking any differences between `pvl` and `faecal` sperm.

One approach would be to calculate the mean `tail.length` per `bird.no` and `sample.type` and perform a paired t-test:

```
mean.tail <- aggregate(s.data$tail.length, by=list(s.data$bird.no, s.data$sample.type), mean)
colnames(mean.tail) <- c("bird.no", "sample.type", "mean.tail")
t.test(mean.tail$mean.tail[mean.tail$sample.type=='pvl'],
       mean.tail$mean.tail[mean.tail$sample.type=='faecal'], paired=TRUE)
```

```
    Paired t-test

data:  mean.tail$mean.tail[mean.tail$sample.type == "pvl"] and mean.tail$mean.tail[mean.tail$sample.typ
t = 0.83982, df = 26, p-value = 0.4087
alternative hypothesis: true mean difference is not equal to 0
95 percent confidence interval:
 -0.6423599  1.5298554
sample estimates:
mean difference
      0.4437478
```

Although the difference in `tail.length` between the two types of sperm is similar to that reported in Table 1, according to our analysis it is still far from significant.

EXERCISE 5.5 (HARD)
Can you perform a paired t-test using `t.test()`, but without specifying `paired=TRUE`?

3.2 Linear model

Our paired t-test tests if the within-male difference between faecal and PVL sperm is different from zero. Although this is an effective way of removing any between-male variation (as each male acts as its own control), it does reduce our dataset from 813 to 27 data points.

> **EXERCISE 5.6**
> What do you think is the correct sample size for our analysis? Is it the number of males, or is it the number of sperm? Compare this analysis to the feeding experiment discussed during the lecture. What are the similarities, and what are the differences?

Rather than removing any variation among males by doing our analyses on mean `tail.length`, at first sight it might seem like a good idea to account for this variation by including `bird.no` as an additional effect in our model:

```
m <- lm(tail.length ~ sample.type + bird.no, data=s.data)
```

> **EXERCISE 5.7**
> Have a look at the output from this model. Is this what you expected? What does the estimate for `bird.no` tell you?

Let's instead fit the same model, but this time with `bird.no` as a factor:

```
m <- lm(tail.length ~ sample.type + as.factor(bird.no), data=s.data)
summary(m)
```

```
Call:
lm(formula = tail.length ~ sample.type + as.factor(bird.no),
    data = s.data)

Residuals:
     Min       1Q   Median       3Q      Max
-13.1393  -0.9022   0.1388   0.9388  14.6850

Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)          22.5744     0.4790  47.132  < 2e-16 ***
sample.typepvl        0.4372     0.1806   2.421 0.015684 *
as.factor(bird.no)2   2.3828     0.5908   4.033 6.04e-05 ***
as.factor(bird.no)3   1.3137     0.6972   1.884 0.059895 .
as.factor(bird.no)4   2.3731     0.6470   3.668 0.000261 ***
as.factor(bird.no)5  16.3890     0.7069  23.185  < 2e-16 ***
as.factor(bird.no)6  20.2780     0.6883  29.461  < 2e-16 ***
as.factor(bird.no)7  -3.7332     0.6366  -5.864 6.65e-09 ***
as.factor(bird.no)8   8.8948     0.5699  15.607  < 2e-16 ***
as.factor(bird.no)9  -0.4072     0.6972  -0.584 0.559318
as.factor(bird.no)10  6.0006     0.6470   9.274  < 2e-16 ***
as.factor(bird.no)11 -6.9060     0.7069  -9.770  < 2e-16 ***
as.factor(bird.no)12  6.3221     0.6231  10.146  < 2e-16 ***
as.factor(bird.no)13 -0.3950     0.6417  -0.616 0.538346
as.factor(bird.no)14 -1.1383     0.6883  -1.654 0.098557 .
as.factor(bird.no)15  5.1064     0.6019   8.483  < 2e-16 ***
as.factor(bird.no)16 14.7247     0.6725  21.896  < 2e-16 ***
as.factor(bird.no)17  7.3582     0.5859  12.558  < 2e-16 ***
as.factor(bird.no)18 13.8533     0.6528  21.223  < 2e-16 ***
as.factor(bird.no)19  4.2240     0.5908   7.149 2.00e-12 ***
```

```
as.factor(bird.no)20  10.7623     0.6589  16.334  < 2e-16 ***
as.factor(bird.no)21  -4.4890     0.6470  -6.938 8.33e-12 ***
as.factor(bird.no)22   2.8721     0.6050   4.747 2.45e-06 ***
as.factor(bird.no)23  14.3360     0.6725  21.318  < 2e-16 ***
as.factor(bird.no)24  -4.3888     0.6883  -6.376 3.09e-10 ***
as.factor(bird.no)25  14.8528     0.6366  23.331  < 2e-16 ***
as.factor(bird.no)26   7.8480     0.5837  13.446  < 2e-16 ***
as.factor(bird.no)27  10.2205     0.6654  15.359  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.376 on 785 degrees of freedom
Multiple R-squared:  0.883, Adjusted R-squared:  0.879
F-statistic: 219.5 on 27 and 785 DF,  p-value: < 2.2e-16
```

This model has estimated an effect for each level of `bird.no`, and thereby it has estimated the effect of `sample.type` while conditioning on `bird.no`. If you compare the estimate of the effect size for `sample.type` to that from a linear model without `bird.no`, you will find that they are very similar. However, this time the standard error of the estimate is much smaller, and as a consequence the p-value for the difference is now below 0.05.

EXERCISE 5.8
Is there any evidence that the effect of `sample.type` varies among males?

Although this seems to have worked quite well, model `m` has estimated a total of 1 (`intercept`) + 1 (`sample.type`) + 26 (`bird.no`) = 28 parameters, using up as many degrees of freedom. Although in this case we have plenty of degrees of freedom left (785), we are not always in such a luxurious position. Furthermore, in this case we are interested in the variation among males, but not in whether, for example, male 12 has sperm with a longer tail than male 1. Finally, although this model may tell us something about this particular random sample of males, it tells us nothing about our study population as a whole.

It is variables such as these that are much better fitted as a *random* (in contrast to a fixed) effect in a so-called mixed model. One of the main differences is that for a random effect we only estimate a variance component, *i.e.* the variance explained by `bird.no`.

3.3 Linear mixed model

To analyse the same data using a mixed model, we need the function `lmer()`, which is part of the `lme4` package. Install `lme4` if you haven't done so already and load it using `library()`:

```
library(lme4)
```

```
Loading required package: Matrix
```

The syntax of `lmer()` is very similar to that of `lm()` and `glm()`, but in addition to the specification of the fixed effects that we are now very familiar with, we also specify a random effect:

```
mm <-  lmer(tail.length ~ sample.type + (1|bird.no), data=s.data)
```

This model estimates the (fixed) effect of `sample.type` as well as the (random) effect of `bird.no`.

Let's have a look at the output using `summary()`:

```
summary(mm)
```

```
Linear mixed model fit by REML ['lmerMod']
Formula: tail.length ~ sample.type + (1 | bird.no)
   Data: s.data
```

```
REML criterion at convergence: 3862

Scaled residuals:
    Min      1Q  Median      3Q     Max
-5.5277 -0.3713  0.0498  0.3823  6.1823

Random effects:
 Groups   Name        Variance Std.Dev.
 bird.no  (Intercept) 51.852   7.201
 Residual              5.644   2.376
Number of obs: 813, groups:  bird.no, 27

Fixed effects:
               Estimate Std. Error t value
(Intercept)     28.0801     1.3933  20.153
sample.typepvl   0.4375     0.1805   2.423

Correlation of Fixed Effects:
            (Intr)
smpl.typpvl -0.083
```

Although most of this looks rather similar to the output from a model fitted using `lm()`, there are two important differences:

1. We now have an estimate for not only the fixed effects (in this case `sample.type`), but also for the random effect `bird.no`. Whereas when we fitted `bird.no` as a fixed effect using `lm()` we obtained an estimate for each bird, now we only have a single estimate: The variance explained by `bird.no` (note that `Std.Dev` simply provides the square-root of the variance). In this case, the variance in `tail.length` among males is equal to 51.9, whereas the variance left unexplained by `bird.no` (and `sample.type`) is 5.6. The latter is called the `Residual` variance, and it is equal to the variance in `tail.length` within males. In other words, 51.852 / (51.852 + 5.644) = 90.2% of the variation in `tail.length` (after accounting for the effect of `sample.type`) is explained by `bird.no`.
2. Although we still get a standard error and t-value for our fixed effect estimates, we no longer get a p-value. The same is true if we use `anova(mm)`, which returns an F-value, but no p-value. This is because it is not obvious what the correct degrees of freedom is (number of males, number of sperm, or something in between), and we need the degrees of freedom to obtain the p-value that comes with a t- or F-value.

Fortunately, we are still able to compare models fitted with `lmer()` using `anova()`. To obtain the significance of `sample.type`, we fit a model without `sample.type` (so only including a fixed intercept, specified by including `1`) as follows:

```
summary(mm.red <-  lmer(tail.length ~ 1 + (1|bird.no), data=s.data))

Linear mixed model fit by REML ['lmerMod']
Formula: tail.length ~ 1 + (1 | bird.no)
   Data: s.data

REML criterion at convergence: 3866.3

Scaled residuals:
    Min      1Q  Median      3Q     Max
-5.6596 -0.3335  0.0600  0.4042  6.0453

Random effects:
 Groups   Name        Variance Std.Dev.
```

```
 bird.no  (Intercept) 51.869   7.202
 Residual              5.679   2.383
Number of obs: 813, groups:  bird.no, 27

Fixed effects:
            Estimate Std. Error t value
(Intercept)   28.361      1.389   20.42
```

We can now test whether `mm.red` is a significantly less good fit to the data than `mm` using `anova()`, which when applied to two mixed models performs a likelihood-ratio test:

```
anova(mm.red, mm)
```

```
refitting model(s) with ML (instead of REML)
```

```
Data: s.data
Models:
mm.red: tail.length ~ 1 + (1 | bird.no)
mm: tail.length ~ sample.type + (1 | bird.no)
       npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
mm.red    3 3874.8 3888.9 -1934.4   3868.8
mm        4 3870.9 3889.7 -1931.5   3862.9 5.8585  1     0.0155 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Based on this, we conclude that "The tail of sperm that have reached the PVL is on average ($\pm$ s.e.) 0.44 ($\pm$ 0.18) μm longer than sperm collected from faecal samples (χ^2_1=5.86, p=0.016)".

Note that the output of `anova()` mentions that it is `refitting model(s) with ML (instead of REML)`. This is because the models were initially fitted using Restricted Maximum Likelihood (or REML). Although REML provides us with a better estimate for the random effect(s) (*i.e.* the variances), when we compare two models containing different fixed effects, we need to first fit both models with Maximum Likelihood instead (ML). Although `anova()` now does this for you (this wasn't the case in older versions), we can also do this ourselves:

```
mm.ml <- lmer(tail.length ~ sample.type + (1|bird.no), data=s.data, REML = FALSE)
mm.red.ml <- lmer(tail.length ~ 1 + (1|bird.no), data=s.data, REML = FALSE)
anova(mm.red.ml, mm.ml)
```

```
Data: s.data
Models:
mm.red.ml: tail.length ~ 1 + (1 | bird.no)
mm.ml: tail.length ~ sample.type + (1 | bird.no)
          npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
mm.red.ml    3 3874.8 3888.9 -1934.4   3868.8
mm.ml        4 3870.9 3889.7 -1931.5   3862.9 5.8585  1     0.0155 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Note that this has produced exactly the same output, but without the warning.

> **EXERCISE 5.9 (VERY HARD)**
> Compare the output of `summary(mm)` and `summary(mm.ml)`. What are the differences and similarities? Can you explain the difference based on what you know about the difference between ML and REML?

Given the debate that surrounds the use of p-values obtained from mixed models, we might want to use another method of assessing statistical significance. For example, a popular approach is to use resampling

techniques (bootstrapping) to obtain a 95%-confidence interval for our estimates, and to check whether this includes 0. This can be done using `confint()`:

```
confint(mm)
```

```
Computing profile confidence intervals ...
```

```
                    2.5 %    97.5 %
.sig01          5.52213107  9.477948
.sigma          2.26145143  2.496534
(Intercept)    25.30303207 30.857264
sample.typepvl  0.08346274  0.791588
```

Finally, if you are willing to make certain assumption regarding how the 'correct' degrees of freedom should be calculated, you could use `KRmodcomp()` in the package `pbkrtest` to use Kenward–Roger approximation to estimate the degrees of freedom:

```
library(pbkrtest)
KRmodcomp(mm.red, mm)
```

```
large : tail.length ~ sample.type + (1 | bird.no)
small : tail.length ~ 1 + (1 | bird.no)
          stat      ndf      ddf F.scaling p.value
Ftest   5.8724   1.0000 785.2243         1 0.01561 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Note that the degrees of freedom are no longer whole numbers!

The package `lmerTest` does something similar but uses Satterthwaite approximation. Once you have loaded this package and you have rerun your model using `lmer()`, `summary(mm)` and `anova(mm)` now return degrees of freedom and p-values:

```
library(lmerTest)
```

```
Attaching package: 'lmerTest'
```

```
The following object is masked from 'package:lme4':

    lmer
```

```
The following object is masked from 'package:stats':

    step
```

```
mm <-  lmer(tail.length ~ sample.type + (1|bird.no), data=s.data)
summary(mm)
```

```
Linear mixed model fit by REML. t-tests use Satterthwaite's method [
lmerModLmerTest]
Formula: tail.length ~ sample.type + (1 | bird.no)
   Data: s.data

REML criterion at convergence: 3862

Scaled residuals:
    Min      1Q  Median      3Q     Max
-5.5277 -0.3713  0.0498  0.3823  6.1823
```

```
Random effects:
 Groups   Name        Variance Std.Dev.
 bird.no  (Intercept) 51.852   7.201
 Residual              5.644   2.376
Number of obs: 813, groups:  bird.no, 27

Fixed effects:
               Estimate Std. Error       df t value Pr(>|t|)
(Intercept)     28.0801     1.3933  26.3285  20.153   <2e-16 ***
sample.typepvl   0.4375     0.1805 785.1910   2.423   0.0156 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
            (Intr)
smpl.typpvl -0.083
```

```
anova(mm)
```

```
Type III Analysis of Variance Table with Satterthwaite's method
            Sum Sq Mean Sq NumDF  DenDF F value Pr(>F)
sample.type 33.143  33.143     1 785.19  5.8725 0.0156 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

If you no longer want to use `lmerTest` and instead you would like the normal `summary()` output back, you will have to detach it again:

```
detach(package:lmerTest)
```

EXERCISE 5.10
Repeat the analyses above for head divided by total sperm length. Discuss a *potential* issue with this analysis, and explore whether this is indeed a problem.

4 Is PVL sperm less variable?

The authors do not only want to test if the measurements of sperm trapped by the outer perivitelline layer (`pvl`) differ from those in the `faeces` in terms of the mean, but they also want to know if they are less variable.

4.1 Calculating and comparing coefficients of variation

To this end they calculate the *coefficient of variation* (CV) for both types of sperm for each male. Remember that the CV of trait x is given by the ratio of the standard deviation (σ_x) to the trait mean ($\bar{x}$), *i.e.* $\frac{\sigma_x}{\bar{x}}$. Often, but not always, this ratio is multiplied by 100 to get a percentage.

EXERCISE 5.11
Why do you think the authors compare coefficients of variation rather than standard deviations? Do you agree?

To calculate the standard deviation for `head.length`, `midpiece.length`, `tail.length` and `total.length` per `bird.no` and `sample.type`, we use `aggregate()`:

```
sd <- aggregate(cbind(s.data$head.length, s.data$midpiece.length,
                      s.data$tail.length, s.data$total.length),
          by=list(s.data$bird.no, s.data$sample.type), sd)
head(sd)
```

```
  Group.1 Group.2        V1       V2       V3       V4
1       1  faecal 0.3699685 1.204392 3.168984 3.377824
2       2  faecal 0.4226491 3.111742 3.266830 3.041563
3       3  faecal 0.2894132 2.641948 2.703640 2.015524
4       4  faecal 0.2195981 2.325240 3.289103 1.918687
5       5  faecal 0.6856700 2.931489 2.293418 1.652546
6       6  faecal 0.4153727 3.640659 3.585204 1.281057
```

We can do the same to calculate the mean for each value of `bird.no` and `sample.type`:

```
mean <- aggregate(cbind(s.data$head.length, s.data$midpiece.length,
                        s.data$tail.length, s.data$total.length),
          by=list(s.data$bird.no, s.data$sample.type), mean)
```

And finally, we divide columns 3 to 6 of `sd` by their respective means and add the columns `bird.no` and `sample.type`:

```
cv <- sd[3:6] / mean[3:6]
cv <- cbind(sd[1:2], cv)
```

Unfortunately `aggregate()` doesn't provide very informative column names, but this we can easily fix using `colnames()`:

```
head(cv)
```

```
  Group.1 Group.2         V1         V2         V3         V4
1       1  faecal 0.03630345 0.04377221 0.15830674 0.05851680
2       2  faecal 0.03698041 0.11051790 0.13086689 0.04712095
3       3  faecal 0.02582663 0.08047359 0.12402016 0.03061432
4       4  faecal 0.02010419 0.07510224 0.14157638 0.02946568
5       5  faecal 0.06788140 0.15247523 0.05723386 0.02381258
6       6  faecal 0.04047283 0.23024656 0.07842855 0.01784500
```

```
colnames(cv) <- c("bird.no", "sample.type", "cv.head", "cv.midpiece", "cv.tail", "cv.total")
head(cv)
```

```
  bird.no sample.type    cv.head cv.midpiece    cv.tail   cv.total
1       1      faecal 0.03630345  0.04377221 0.15830674 0.05851680
2       2      faecal 0.03698041  0.11051790 0.13086689 0.04712095
3       3      faecal 0.02582663  0.08047359 0.12402016 0.03061432
4       4      faecal 0.02010419  0.07510224 0.14157638 0.02946568
5       5      faecal 0.06788140  0.15247523 0.05723386 0.02381258
6       6      faecal 0.04047283  0.23024656 0.07842855 0.01784500
```

Now we are ready to test if the coefficient of variation differs between sample types:

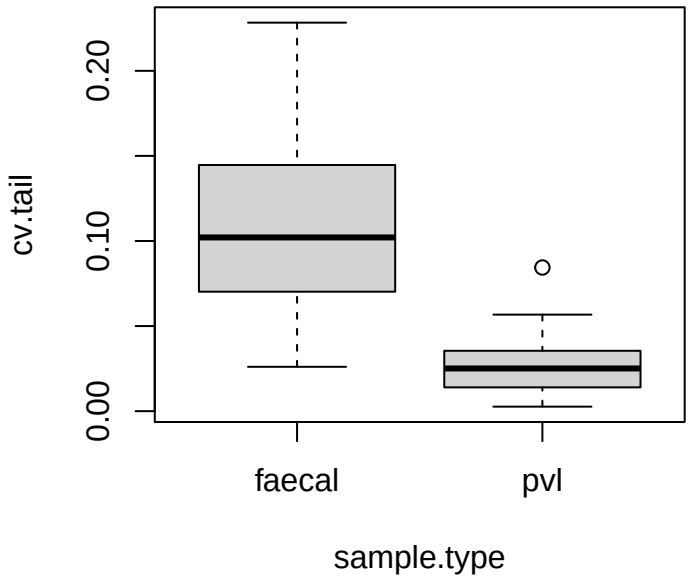
```
t.test(cv$cv.tail[cv$sample.type=='pvl'], cv$cv.tail[cv$sample.type=='faecal'], paired=TRUE)
```

```

    Paired t-test

data:  cv$cv.tail[cv$sample.type == "pvl"] and cv$cv.tail[cv$sample.type == "faecal"]
t = -8.1232, df = 26, p-value = 1.327e-08
alternative hypothesis: true mean difference is not equal to 0
95 percent confidence interval:
 -0.10707323 -0.06382753
sample estimates:
mean difference
    -0.08545038
```

```
plot(cv.tail ~ sample.type, data = cv)
```



In line with the results obtained by the authors (see Figure 1d), this suggests a strong reduction in the variability in `tail.length`.

4.2 A mixed model approach

We often include a random effect because we want to obtain a better estimate for one or more fixed effects. For example, in Section 3 we included `bird.no` to obtain a better estimate of the difference in `tail.length` between sample types, and in the lecture we accounted for the non-independence of nestling that have grown up in the same nest. However, sometimes the random effects themselves are of interest to us as well.

For example, by including `bird.no` as a random effect, we obtained an estimate of the variance among males, as well as the residual variance, with the latter being equal to the variance within males. This means that we could directly estimate the within- and among-male variance in `tail.length` for both sample types by fitting a mixed models to both of the two sample types separately:

```
mm.faecal <- lmer(tail.length ~ 1 + (1|bird.no), data=s.data[s.data$sample.type=="faecal", ])
summary(mm.faecal)
```

```
Linear mixed model fit by REML ['lmerMod']
Formula: tail.length ~ 1 + (1 | bird.no)
   Data: s.data[s.data$sample.type == "faecal", ]

REML criterion at convergence: 1520.4

Scaled residuals:
    Min      1Q  Median      3Q     Max
-3.2851 -0.5654  0.0294  0.5789  4.0682

Random effects:
 Groups   Name        Variance Std.Dev.
 bird.no  (Intercept) 69.12    8.314
 Residual             10.92    3.305
Number of obs: 270, groups:  bird.no, 27

Fixed effects:
```

```
            Estimate Std. Error t value
(Intercept)   28.080      1.613   17.41
```

```
mm.pvl <- lmer(tail.length ~ 1 + (1|bird.no), data=s.data[s.data$sample.type=="pvl", ])
summary(mm.pvl)
```

```
Linear mixed model fit by REML ['lmerMod']
Formula: tail.length ~ 1 + (1 | bird.no)
   Data: s.data[s.data$sample.type == "pvl", ]

REML criterion at convergence: 1768

Scaled residuals:
    Min      1Q  Median      3Q     Max
-4.7009 -0.4477  0.0089  0.4652  3.3552

Random effects:
 Groups   Name        Variance Std.Dev.
 bird.no  (Intercept) 41.798   6.465
 Residual              1.104   1.051
Number of obs: 543, groups:  bird.no, 27

Fixed effects:
            Estimate Std. Error t value
(Intercept)   28.524      1.245   22.91
```

Although they are not identical (and we don't expect them to be identical), they are very similar to the mean within-male standard deviations for both sample types obtained in the previous section:

```
aggregate(sd$V3, by=list(sd$Group.2), mean)
```

```
  Group.1         x
1  faecal 3.0056617
2     pvl 0.8157826
```

Unfortunately, testing if the difference in the within-male variance between sample types is statistically significant is not straightforward using `lme4` (but it can be done in other packages, such as `asreml-r`). However, we could use `confint()` to obtain confidence intervals for both residual variances, and see if these overlap. Because comparing two 95%-confidence intervals would too conservative (Goldstein and Healy 1995), we should instead use an 83% confidence interval:

```
confint(mm.faecal, level = 0.83)
```

```
Computing profile confidence intervals ...
```

```
                8.5 %    91.5 %
.sig01       6.817216  9.979520
.sigma       3.109509  3.521923
(Intercept) 25.870338 30.289958
```

```
confint(mm.pvl, level = 0.83)
```

```
Computing profile confidence intervals ...
```

```
                8.5 %    91.5 %
.sig01       5.319011  7.744277
.sigma       1.007215  1.097065
(Intercept) 26.817648 30.230383
```

Although we are now comparing variances rather coefficients of variation, we again find that PVL sperm is less variable than faecal sperm.

EXERCISE 5.12 (VERY HARD)
Would it be possible to directly estimate coefficients of variation using a mixed model?