Random Effects (I)

Jarrod Hadfield

University of Edinburgh

Random Effects (I)

Jarrod Hadfield

University of Edinburgh

So far the coefficients in our (generalised) linear model have been what are called fixed effects and in this lecture we are going to introduce another set of coefficients that are called random effects. Random effects generally confuse people so we'll go through it slowly, and for those that have already met them, I will try and dispel some of the misinformation you've almost certainly been given.

Linear Model

- Model Syntax for Fixed Effects

```
y ~ limit + year + day
```


Linear Model
- Model Syntax for Fixed Effects
y ~ limit + year + day

First, let's briefly go back and think about how we specified a linear model, in this case for the model where we analysed the number of accidents on Swedish roads. We had the number of accidents as a function of whether a speed limit was in place or not, which year it was (1961 or 1962) and day: note we are treating day as a continuous variable here: we're asking is there a change in the number of accidents throughout the year - from January to December - and the regression coefficient is in units of accidents *per day*.

Linear Model

- Model Syntax for Fixed Effects

```
y ~ limit + year + day
```

- Set of Simultaneous Equations

$$
\begin{aligned}
E[\texttt{y[1]}] &= 1\beta_1 + (\texttt{limit[1]=="yes"})\beta_2 + (\texttt{year[1]=="1962"})\beta_3 + \texttt{day[1]}\beta_4 \\
E[\texttt{y[2]}] &= 1\beta_1 + (\texttt{limit[2]=="yes"})\beta_2 + (\texttt{year[2]=="1962"})\beta_3 + \texttt{day[2]}\beta_4 \\
\vdots &= \vdots \\
E[\texttt{y[184]}] &= 1\beta_1 + (\texttt{limit[184]=="yes"})\beta_2 + (\texttt{year[184]=="1962"})\beta_3 + \texttt{day[184]}\beta_4
\end{aligned}
$$

Linear Model
Model Syntax for Fixed Effects
y ~ limit + year + day
Set of Simultaneous Equations

We saw that the model syntax is setting up a large set of simultaneous equations. On the left we have the expected number of accidents given the information on the right. In blue we have data that we've gone out and collected and in red we have the parameters we'd like to estimate using that data. On the right hand side we have an intercept of all ones, continuous predictors such as day which remain unchanged and categorical predictors such as year and speed limit that are expanded into a series of binary variables of the form 'do the data come from days with a speed limit, yes or no?', 'do the data come from 1962, yes or no?'. And the key thing is that although you can do what you want with the data (the blue things) you are never multiplying or dividing the parameters, the things in red, by each other. That's what makes a linear model linear.

Now we've only looked at these equations for three data points and it's already starting to look pretty overwhelming.

Linear Model

- Model Syntax for Fixed Effects

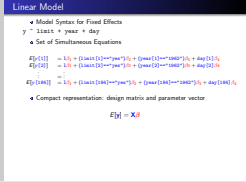
```
y ~ limit + year + day
```

- Set of Simultaneous Equations

$$
\begin{aligned}
E[y\texttt{[1]}] &= 1\beta_1 + (\texttt{limit[1]=="yes"})\beta_2 + (\texttt{year[1]=="1962"})\beta_3 + \texttt{day[1]}\beta_4 \\
E[y\texttt{[2]}] &= 1\beta_1 + (\texttt{limit[2]=="yes"})\beta_2 + (\texttt{year[2]=="1962"})\beta_3 + \texttt{day[2]}\beta_4 \\
\vdots &= \vdots \\
E[y\texttt{[184]}] &= 1\beta_1 + (\texttt{limit[184]=="yes"})\beta_2 + (\texttt{year[184]=="1962"})\beta_3 + \texttt{day[184]}\beta_4
\end{aligned}
$$

- Compact representation: design matrix and parameter vector

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



However, was saw that we could represent this whole system of equations very compactly in terms of matrices and vectors. The $\mathbf{X}$ matrix we call a design matrix and each column contains a predictor which is associated with an element of our parameter vector, β. In this example, $\mathbf{X}$ is a 184×4 matrix (traditionally the number of rows is reported before the number of columns) and the parameter vector is a 4×1 matrix (i.e. a vector of length 4). When we multiply a matrix by a vector we are carrying out the operation highlighted above: we're taking each row in the matrix, multiplying each element by the corresponding element in the vector, and then adding them up.

Linear Model

- Model Syntax for Fixed Effects

```
y ~ limit + year + day
```

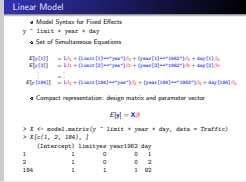
- Set of Simultaneous Equations

$$
\begin{aligned}
E[y[1]] &= 1\beta_1 + (\texttt{limit[1]=="yes"})\beta_2 + (\texttt{year[1]=="1962"})\beta_3 + \texttt{day[1]}\beta_4 \\
E[y[2]] &= 1\beta_1 + (\texttt{limit[2]=="yes"})\beta_2 + (\texttt{year[2]=="1962"})\beta_3 + \texttt{day[2]}\beta_4 \\
&\vdots \quad = \quad \vdots \\
E[y[184]] &= 1\beta_1 + (\texttt{limit[184]=="yes"})\beta_2 + (\texttt{year[184]=="1962"})\beta_3 + \texttt{day[184]}\beta_4
\end{aligned}
$$

- Compact representation: design matrix and parameter vector

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

```
> X <- model.matrix(y ~ limit + year + day, data = Traffic)
> X[c(1, 2, 184), ]
    (Intercept) limityes year1962 day
1             1        0        0   1
2             1        0        0   2
184           1        1        1  92
```

If you are uncertain about what model you have specified, a useful thing to do is to have a look at the design matrix using the function `model.matrix`. We can see that the 184th observation was made on day 92 of 1962 and yes there was a speed limit. If we take each number and multiply it by its corresponding parameter we can take the sum of these products to get our prediction for the expected number of accidents ($1\beta_1 + 1\beta_2 + 1\beta_3 + 92\beta_4$)

Linear Model

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

Linear Model

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

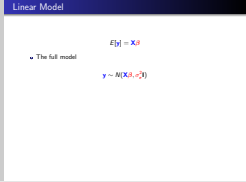
OK- so we've got our set of simultaneous equations and the problem is that we can't just solve them because we don't actually know the quantity on the left hand side. The quantity on the left hand side is the *expected* number of accidents given our model but we don't know the expectation for each row in our data frame, we just have the single realisation that we've observed.

Linear Model

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

- The full model

$$\mathbf{y} \sim N(\mathbf{X}\beta, \sigma_e^2\mathbf{I})$$



So we have to do a bit more work. We have to specify how the actual number of accidents will deviate from the expected value. In a standard linear model we assume that the deviations around the expected value (the residuals) are normally distributed, and that the variance of these deviations is to be estimated.

Linear Model

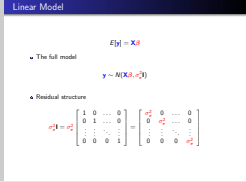
$$E[\mathbf{y}] = \mathbf{X}\beta$$

- The full model

$$\mathbf{y} \sim N(\mathbf{X}\beta, \sigma_e^2\mathbf{I})$$

- Residual structure

$$\sigma_e^2\mathbf{I} = \sigma_e^2 \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \sigma_e^2 & 0 & \dots & 0 \\ 0 & \sigma_e^2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \sigma_e^2 \end{bmatrix}$$



The standard assumption about these deviations is that they are identically and independently distributed. This is a 184×184 matrix. The diagonals specify how much variance around the expectation we expect for each observation and we assume this is constant (the deviations are identically distributed). This doesn't mean that each observation will lie equally far from its expectation, it means that each observation has the same *probability* of lying equally far from its expectation. The off-diagonals are zero and this implies that the deviations are uncorrelated. This means that if one observation happens to be above its expectation then this doesn't tell you anything about the deviations of other observations.

Linear Mixed Model

- Model Syntax for Random Effects

```
y ~ as.factor(day)-1
```


Linear Mixed Model
Model Syntax for Random Effects
y ~ as.factor(day)-1

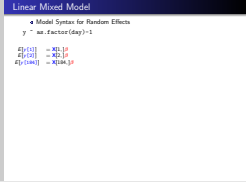
Now we're going to add day effects into the model, and we're going to treat them as random. Note that here we are treating day as a categorical variable rather than a continuous covariate, and I've also removed the intercept which I'll come back to.

Linear Mixed Model

- Model Syntax for Random Effects

```
y ~ as.factor(day)-1
```

$$
\begin{aligned}
E[y\texttt{[1]}] &= \mathbf{X}[1,]\beta \\
E[y\texttt{[2]}] &= \mathbf{X}[2,]\beta \\
E[y\texttt{[184]}] &= \mathbf{X}[184,]\beta
\end{aligned}
$$



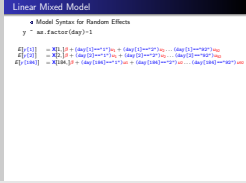
Here we have the model so far (the fixed effect part), where the expected number of accidents for the 1st observation is the predictor data (intercept, year, speed-limit, day *as a number*) for that observation (the 1st row of the design matrix) multiplied by the parameter vector. What we're then going to do is add day effects to this model.

Linear Mixed Model

- Model Syntax for Random Effects

```
y ~ as.factor(day)-1
```

$$
\begin{aligned}
E[y\texttt{[1]}] &= \mathbf{X}[1,]\beta + \texttt{(day[1]=="1")}u_1 + \texttt{(day[1]=="2")}u_2 \ldots \texttt{(day[1]=="92")}u_{92} \\
E[y\texttt{[2]}] &= \mathbf{X}[2,]\beta + \texttt{(day[2]=="1")}u_1 + \texttt{(day[2]=="2")}u_2 \ldots \texttt{(day[2]=="92")}u_{92} \\
E[y\texttt{[184]}] &= \mathbf{X}[184,]\beta + \texttt{(day[184]=="1")}u_1 + \texttt{(day[184]=="2")}u_2 \ldots \texttt{(day[184]=="92")}u_{92}
\end{aligned}
$$



And they're added in exactly the same way, we have our data in blue, and each column contains either ones or zeros: 'is this day 1, yes or no?', 'is this day 2, yes or no?' and so on. And each of these predictor variables is going to be multiplied by a coefficient that corresponds to the effect of day 1 (u_1), and day 2 (u_2) and so on all the way up to day 92 (u_{92}). In this particular example, each observation is only associated with a single day, so for any row the new predictor variables are going to be all zeros except for one. So for example, observation 1 was made on day 1 and so only `(day[1]=="1")` is equal to 1 and so only u_1 is added to the prediction.

Linear Mixed Model

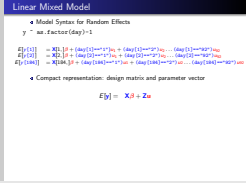
- Model Syntax for Random Effects

```
y ~ as.factor(day)-1
```

$$
\begin{aligned}
E[y\texttt{[1]}] &= \mathbf{X}[1,]\beta + (\texttt{day[1]=="1"})u_1 + (\texttt{day[1]=="2"})u_2 \ldots (\texttt{day[1]=="92"})u_{92} \\
E[y\texttt{[2]}] &= \mathbf{X}[2,]\beta + (\texttt{day[2]=="1"})u_1 + (\texttt{day[2]=="2"})u_2 \ldots (\texttt{day[2]=="92"})u_{92} \\
E[y\texttt{[184]}] &= \mathbf{X}[184,]\beta + (\texttt{day[184]=="1"})u_1 + (\texttt{day[184]=="2"})u_2 \ldots (\texttt{day[184]=="92"})u_{92}
\end{aligned}
$$

- Compact representation: design matrix and parameter vector

$$E[\mathbf{y}] = \mathbf{X}\beta + \mathbf{Z}\mathbf{u}$$



Often the design matrix for the random effects is denoted by a **Z** (rather than by an **X**) and the parameter vector by **u** rather than by β.

Linear Mixed Model

- Model Syntax for Random Effects

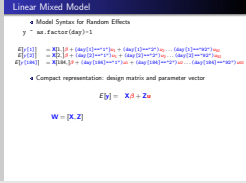
```
y ~ as.factor(day)-1
```

$$
\begin{aligned}
E[y\texttt{[1]}] &= \mathbf{X}[1,]\beta + \texttt{(day[1]=="1")}u_1 + \texttt{(day[1]=="2")}u_2 \ldots \texttt{(day[1]=="92")}u_{92} \\
E[y\texttt{[2]}] &= \mathbf{X}[2,]\beta + \texttt{(day[2]=="1")}u_1 + \texttt{(day[2]=="2")}u_2 \ldots \texttt{(day[2]=="92")}u_{92} \\
E[y\texttt{[184]}] &= \mathbf{X}[184,]\beta + \texttt{(day[184]=="1")}u_1 + \texttt{(day[184]=="2")}u_2 \ldots \texttt{(day[184]=="92")}u_{92}
\end{aligned}
$$

- Compact representation: design matrix and parameter vector

$$E[\mathbf{y}] = \mathbf{X}\beta + \mathbf{Z}\mathbf{u}$$

$$\mathbf{W} = [\mathbf{X}, \mathbf{Z}]$$



But in some ways this overemphasises the difference, we could just combine the columns of **X** and **Z** and define a new design matrix **W** (irrespective of whether it contains predictors for fixed effects, random effects or both).

Linear Mixed Model

- Model Syntax for Random Effects

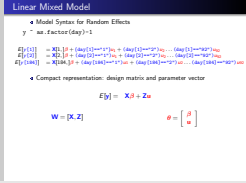
```
y ~ as.factor(day)-1
```

$$
\begin{aligned}
E[y\texttt{[1]}] &= \mathbf{X}[1,]\beta + (\texttt{day[1]=="1")}u_1 + (\texttt{day[1]=="2")}u_2 \ldots (\texttt{day[1]=="92")}u_{92} \\
E[y\texttt{[2]}] &= \mathbf{X}[2,]\beta + (\texttt{day[2]=="1")}u_1 + (\texttt{day[2]=="2")}u_2 \ldots (\texttt{day[2]=="92")}u_{92} \\
E[y\texttt{[184]}] &= \mathbf{X}[184,]\beta + (\texttt{day[184]=="1")}u_1 + (\texttt{day[184]=="2")}u_2 \ldots (\texttt{day[184]=="92")}u_{92}
\end{aligned}
$$

- Compact representation: design matrix and parameter vector

$$E[\mathbf{y}] = \mathbf{X}\beta + \mathbf{Z}\mathbf{u}$$

$$\mathbf{W} = [\mathbf{X}, \mathbf{Z}] \qquad \theta = \begin{bmatrix} \beta \\ \mathbf{u} \end{bmatrix}$$



And we could also group the parameters into a single vector, $\boldsymbol{\theta}$; we just stack the fixed and random effects on top of each other.

Linear Mixed Model

- Model Syntax for Random Effects

```
y ~ as.factor(day)-1
```

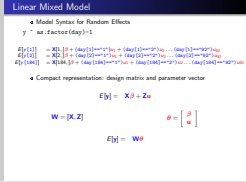
$$
\begin{aligned}
E[y\texttt{[1]}] &= \mathbf{X}[1,]\beta + (\texttt{day[1]=="1")}u_1 + (\texttt{day[1]=="2")}u_2 \ldots (\texttt{day[1]=="92")}u_{92} \\
E[y\texttt{[2]}] &= \mathbf{X}[2,]\beta + (\texttt{day[2]=="1")}u_1 + (\texttt{day[2]=="2")}u_2 \ldots (\texttt{day[2]=="92")}u_{92} \\
E[y\texttt{[184]}] &= \mathbf{X}[184,]\beta + (\texttt{day[184]=="1")}u_1 + (\texttt{day[184]=="2")}u_2 \ldots (\texttt{day[184]=="92")}u_{92}
\end{aligned}
$$

- Compact representation: design matrix and parameter vector

$$E[\mathbf{y}] = \mathbf{X}\beta + \mathbf{Z}\mathbf{u}$$

$$\mathbf{W} = [\mathbf{X}, \mathbf{Z}] \qquad \theta = \begin{bmatrix} \beta \\ \mathbf{u} \end{bmatrix}$$

$$E[\mathbf{y}] = \mathbf{W}\theta$$



And what we end up with is a single set of simultaneous equations as before.

Linear Mixed Model

- Model Syntax for Random Effects

```
y ~ as.factor(day)-1
```

$$
\begin{aligned}
E[y\texttt{[1]}] &= \mathbf{X}[1,]\beta + (\texttt{day[1]=="1"})u_1 + (\texttt{day[1]=="2"})u_2 \ldots (\texttt{day[1]=="92"})u_{92} \\
E[y\texttt{[2]}] &= \mathbf{X}[2,]\beta + (\texttt{day[2]=="1"})u_1 + (\texttt{day[2]=="2"})u_2 \ldots (\texttt{day[2]=="92"})u_{92} \\
E[y\texttt{[184]}] &= \mathbf{X}[184,]\beta + (\texttt{day[184]=="1"})u_1 + (\texttt{day[184]=="2"})u_2 \ldots (\texttt{day[184]=="92"})u_{92}
\end{aligned}
$$

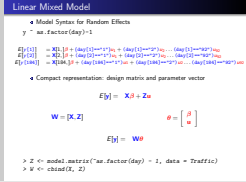
- Compact representation: design matrix and parameter vector

$$E[\mathbf{y}] = \mathbf{X}\beta + \mathbf{Z}\mathbf{u}$$

$$\mathbf{W} = [\mathbf{X}, \mathbf{Z}] \qquad \theta = \begin{bmatrix} \beta \\ \mathbf{u} \end{bmatrix}$$

$$E[\mathbf{y}] = \mathbf{W}\theta$$

```
> Z <- model.matrix(~as.factor(day) - 1, data = Traffic)
> W <- cbind(X, Z)
```

And again you can make these matrices in R easily using the `model.matrix` function: you have to make sure day is treated as a factor and you have to remove the intercept - you don't want a column of ones in the first column because an intercept term has already been fitted.

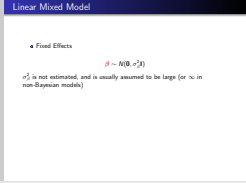
So if we don't need to distinguish between fixed and random effects at this stage, when should we distinguish between them, and what distinguishes them? Often the distinction between fixed and random is given by example; things like population, species, individual and vial are random, but sex, treatment and age are not. Or the distinction is made using rules of thumb; if there are few factor levels and they are interesting to other people they are fixed. However, this doesn't really confer any understanding about what it means to treat something as fixed or random, and doesn't really allow judgements to be made regarding ambiguous variables (for example year) or give any insight into the fact that in a Bayesian analysis all effects are technically random.

Linear Mixed Model

- Fixed Effects

$$\beta \sim N(\mathbf{0}, \sigma^2_\beta \mathbf{I})$$

σ^2_β is not estimated, and is usually assumed to be large (or ∞ in non-Bayesian models)



So in a nutshell - this is the difference. When we treat something as fixed we believe that the parameters have some distribution: often we assume they are normal, we generally assume that they are independent (represented by this identity matrix) and we generally assume that the variance of this distribution is large (in fact infinity in non-Bayesian analyses [1]). We don't estimate this variance we assume it is something large. And when the variance is large we are basically saying that the true value of the parameter could lie anywhere - it is almost as likely to be a thousand as it is 0.

[1] Statisticians won't like this definition. When the variance of the Gaussian is infinite the resulting distribution is not really a distribution any more (it's improper) and so technically you wouldn't think of the fixed effects as coming from a distribution. This is where the name fixed, as opposed to random, comes from - they are not random variables with a specified distribution. However, I think this is a nice way of highlighting the practical difference between the two types of effect, so lets pretend that in a pure likelihood framework that variance isn't really infinite but something very very large.

Linear Mixed Model

- Fixed Effects

$$\beta \sim N(\mathbf{0}, \sigma^2_\beta \mathbf{I})$$

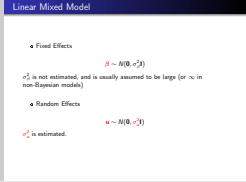
σ^2_β is not estimated, and is usually assumed to be large (or ∞ in non-Bayesian models)

- Random Effects

$$\mathbf{u} \sim N(\mathbf{0}, \sigma^2_u \mathbf{I})$$

σ^2_u is estimated.

Random Effects (I)



Random effects differ by one important point - rather than assuming that the variance is large (or infinite) we actually estimate the variance from the data. The variance may be small - it may be zero for example - in which case all the random effects are going to be forced to zero, or it could be very large and the random effects behave more like fixed effects. This is the difference: for fixed effects we assume a very large or even infinite variance, for random effects we estimate a finite variance. The difference really is that simple, but it takes a long time and a lot of practice to understand what this means in practical terms, and why working with random effects can be a very powerful way of modelling data. When I explain random effects in this way to people that have been using random effects for some time they love it; things fall into place. But I'm aware that for those of you that have little experience with mixed models this distinction seems very abstract. So lets try and see why it's useful in less abstract terms.

Imagine we went out and we measured the sex ratio of some broods of great tits - and lets imagine that the first 10 broods were massive: each had a thousand offspring and that the sex ratio in each case was close to 50:50 (there will still be bit of noise of course because even if the expected sex ratio was 50:50 we would expect some deviations around this just by chance). Now lets say we went to the 11th brood and there was only a single chick and he was male. The sex ratio in this brood is 1. Now if we were to treat these brood effects as fixed our best estimates would be 0.51, 0.52, 0.48 ... and then 1: now who here would think that if this bird had more offspring they would all be male. Nobody I think: you would use the knowledge that you have gained from these other nests and say that you've got really accurate measures of sex ratio for these 10 broods and you would say each one only deviates from 50:50 slightly and so most likely if you had collected more data on the 11th brood it would lie in that range. If you had collected a thousand offspring form this nest and they'd all had been male: fair enough you would be more willing to say that the expected sex ratio in this nest is 100% and the data collected on that brood would outweigh the prior information you gain from the other nests. So when you treat something as random you use the information that you've gathered on that nest (so what sex ratio have we observed) and weight this by what we expect given what we observe in the other nests. And the amount of weighting we do for a particular nest depends on how much data we have for it: and how big the variance is. If the variance is small it implies that the variability in these nest effects is small and so they are useful for predicting another observation. Alternatively if the variance is large they don't offer much predictive power. Imagine that we had still collected a 1000 offspring from each of these broods but the sex ratios had been 0.9, 0.5, 0.01 and so on. The average is still 0.5 but you would be much less confident that if you had sampled more chicks from this nest they would turn out to be half males and half females: that might be so, but perhaps this nest is more like the first one and the rest of the chicks will all be male, or maybe it is more like the third one and the majority are going to turn out female. In this case these other nests provide little information about the 11th nest, and so we would say that our best estimate is a sex ratio close to one, *but* there's a lot of uncertainty in this value. In the previous example we still only had one observation for the 11th nest but we could be fairly confident that the expected sex-ratio was close to 50:50. Now of course, this is an odd scenario; we have 10 nests in which we have so much information we know the sex-ratio in each nest effect almost perfectly, and then we can use the variability in these nest effects to inform us what the likely spread of sex-ratios is likely to be in future nests such as the 11th nest. In reality we usually have nests where the number of offspring is modest for all nests and so all underlying nest effects are not know with much certainty. But the logic is the same; there is still *some* information about each underlying nest effect, and we can use this information to say how variable the true nest effects are likely to be.

Linear Mixed Model

Fixed Effects

$\boldsymbol{\beta} \sim N(\mathbf{0}, \sigma^2_\beta \mathbf{I})$

σ^2_β not estimated (∞).

Random Effects

$\mathbf{u} \sim N(\mathbf{0}, \sigma^2_u \mathbf{I})$

σ^2_u estimated.

Linear Mixed Model

Fixed Effects

$\boldsymbol{\beta} \sim N(\mathbf{0}, \sigma^2_\beta \mathbf{I})$

σ^2_β not estimated (∞).

Random Effects

$\mathbf{u} \sim N(\mathbf{0}, \sigma^2_u \mathbf{I})$

σ^2_u estimated.

So, when we treat effects as fixed only data associated with that fixed effect is used to obtain an estimate. The only information used to estimate nest 1's effect is the number of males and females in nest 1.

When we treat effects as random, the number of males and females in nest 1 is also used to estimate nest 1's effect, but we also use the data associated with all other nests to estimate a variance (σ^2_u) which puts bounds on what are plausible values for nest effects. If the data collected on nest 1 suggest nest 1's nest effect is very extreme then we might put this down to a small sample size within nest 1 and shift the nest effect towards the global mean. This is why random effect estimation is also known as shrinkage, and the degree to which we shrink depends on how variable the underlying nest effects are.

Linear Mixed Model

Fixed Effects	Random Effects
$\boldsymbol{\beta} \sim N(\mathbf{0}, \sigma^2_\beta \mathbf{I})$	$\mathbf{u} \sim N(\mathbf{0}, \sigma^2_u \mathbf{I})$
σ^2_β not estimated (∞).	σ^2_u estimated.

- Random effects are called random because they come from a distribution.

Linear Mixed Model

Fixed Effects Random Effects

$\boldsymbol{\beta} \sim N(\mathbf{0}, \sigma^2_\beta \mathbf{I})$ $\mathbf{u} \sim N(\mathbf{0}, \sigma^2_u \mathbf{I})$

σ^2_β not estimated (∞). σ^2_u estimated.

- Random effects are called random because they come from a distribution.

This is why random effects are called random: the effects are random because they are assumed to come from a distribution. It has nothing to do with whether nests are sampled at random or not. Certainly, if they weren't sampled at random we may question whether all nest effects belong to the same distribution, but the key thing is that we would still treat the nest effects as coming from a distribution and so the nest effects are random.

Linear Mixed Model

Fixed Effects	Random Effects
$\boldsymbol{\beta} \sim N(\mathbf{0}, \sigma^2_\beta \mathbf{I})$	$\mathbf{u} \sim N(\mathbf{0}, \sigma^2_u \mathbf{I})$
σ^2_β not estimated (∞).	σ^2_u estimated.

- Random effects are called random because they come from a distribution.
- Having $\sigma^2_\beta = \infty$ (pure likelihood) is equivalent to saying there is no distribution, hence fixed effects.

Linear Mixed Model

Fixed Effects

$\boldsymbol{\beta} \sim N(\mathbf{0}, \sigma^2_\beta \mathbf{I})$

σ^2_β not estimated (∞).

Random Effects

$\mathbf{u} \sim N(\mathbf{0}, \sigma^2_u \mathbf{I})$

σ^2_u estimated.

- Random effects are called random because they come from a distribution.
- Having $\sigma^2_\beta = \infty$ (pure likelihood) is equivalent to saying there is no distribution, hence fixed effects.

In the pure likelihood approaches we've been using and will continue to use, the variance of the fixed effects is effectively set at infinity in which case we don't really have a distribution and that is why the effects are referred to as fixed.

Linear Mixed Model

Fixed Effects	Random Effects
$\boldsymbol{\beta} \sim N(\mathbf{0}, \sigma^2_\beta \mathbf{I})$	$\mathbf{u} \sim N(\mathbf{0}, \sigma^2_u \mathbf{I})$
σ^2_β not estimated (∞).	σ^2_u estimated.

- Random effects are called random because they come from a distribution.
- Having $\sigma^2_\beta = \infty$ (pure likelihood) is equivalent to saying there is no distribution, hence fixed effects.
- Having $\sigma^2_\beta \neq \infty$ (Bayesian) the 'fixed' effects are also random (but useful to keep the terminology)

Linear Mixed Model

Fixed Effects

$\boldsymbol{\beta} \sim N(\mathbf{0}, \sigma^2_\beta \mathbf{I})$

σ^2_β not estimated (∞).

Random Effects

$\mathbf{u} \sim N(\mathbf{0}, \sigma^2_u \mathbf{I})$

σ^2_u estimated.

- Random effects are called random because they come from a distribution.
- Having $\sigma^2_\beta = \infty$ (pure likelihood) is equivalent to saying there is no distribution, hence fixed effects.
- Having $\sigma^2_\beta \neq \infty$ (Bayesian) the 'fixed' effects are also random (but useful to keep the terminology)

In a Bayesian analysis you usually set the variance, and possibly the mean, to reflect your prior information about the magnitude of the effect. For example, if the effect was the difference in height between mean and women you probably have a feel for what is plausible before you've measured people, for example the people in this room. You might think the average difference is probably around 15 cm and certainly greater than 5 cm and less than 25cm. You could then specify a variance that was consistent with this prior information. In practice, people often do not use highly-informative priors, but use large variances as a way of saying they are uncertain about the true value. But the variance is still finite so a distribution is specified and so technically the 'fixed' effects in a Bayesian analysis are also random. However, I think it is still useful to call them fixed because the terminology is well understood; these are the effects for which the variance is specified a priori (usually at a high value) rather than estimated.

Linear Mixed Model

Fixed Effects	Random Effects
$\beta \sim N(\mathbf{0}, \sigma^2_\beta \mathbf{I})$	$\mathbf{u} \sim N(\mathbf{0}, \sigma^2_u \mathbf{I})$
σ^2_β not estimated (∞).	σ^2_u estimated.

- Random effects are called random because they come from a distribution.
- Having $\sigma^2_\beta = \infty$ (pure likelihood) is equivalent to saying there is no distribution, hence fixed effects.
- Having $\sigma^2_\beta \neq \infty$ (Bayesian) the 'fixed' effects are also random (but useful to keep the terminology)

The wrong sentence

- Nest was treated as random

Linear Mixed Model

Fixed Effects

$\beta \sim N(\mathbf{0}, \sigma^2_\beta \mathbf{I})$

σ^2_β not estimated (∞).

Random Effects

$\mathbf{u} \sim N(\mathbf{0}, \sigma^2_u \mathbf{I})$

σ^2_u estimated.

- Random effects are called random because they come from a distribution.
- Having $\sigma^2_\beta = \infty$ (pure likelihood) is equivalent to saying there is no distribution, hence fixed effects.
- Having $\sigma^2_\beta \neq \infty$ (Bayesian) the 'fixed' effects are also random (but useful to keep the terminology)

The wrong sentence

- Nest was treated as random

A tremendous amount of confusion and misunderstanding surrounds random effects and you see missinformation everywhere in books, papers, help-lists and talks. I think this type of sentence embodies the confusion. *'Nest was treated as random'* or *'Nest was treated as a random effect'* It seems to imply that a) nests are somehow chosen at random and that b) only a single effect has been estimated.

Linear Mixed Model

Fixed Effects	Random Effects
$\boldsymbol{\beta} \sim N(\mathbf{0}, \sigma^2_\beta \mathbf{I})$	$\mathbf{u} \sim N(\mathbf{0}, \sigma^2_u \mathbf{I})$
σ^2_β not estimated (∞).	σ^2_u estimated.

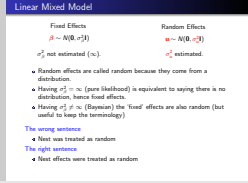
- Random effects are called random because they come from a distribution.
- Having $\sigma^2_\beta = \infty$ (pure likelihood) is equivalent to saying there is no distribution, hence fixed effects.
- Having $\sigma^2_\beta \neq \infty$ (Bayesian) the 'fixed' effects are also random (but useful to keep the terminology)

The wrong sentence

- Nest was treated as random

The right sentence

- Nest effects were treated as random



Much better to say *'Nest effects were treated as random'* This makes it clear that it is the effects that are random (they are assumed to come from a distribution) not nests, and that there are multiple effects (one for each nest) rather than just a single effect. It is true that we are estimating a single variance parameter for the distribution of nest effects but there are multiple nest effects. If there wasn't it would be hard to estimate their variance!

Linear Mixed Model

- Residual structure

$$\sigma_e^2 \mathbf{I} = \sigma_e^2 = \begin{bmatrix} \sigma_e^2 & 0 & 0 & \dots & 0 \\ 0 & \sigma_e^2 & 0 & \dots & 0 \\ 0 & 0 & \sigma_e^2 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \sigma_e^2 \end{bmatrix}$$

Linear Mixed Model

- Residual structure

That is one way to think about how random effects work and for many it will be the most useful way to think about them. There is another way to think about them though that might be useful for others so I'll go through it. Previously I represented the residual structure of our model like this: we have an n by n matrix where n is the number of observations, and we specified that our residuals were independent and identically distributed. The same residual variance is along the diagonal which represents our assumption that the noise around each expectation is equal. And I stress here that when I say noise I mean both measurement error (if there is any) but also biological noise; the effects of all the relevant bits of the world that effect your observations but you haven't measured.

Linear Mixed Model

- Residual structure

$$\sigma_e^2 \mathbf{I} = \sigma_e^2 = \begin{bmatrix} \sigma_e^2 & 0 & 0 & \dots & 0 \\ 0 & \sigma_e^2 & 0 & \dots & 0 \\ 0 & 0 & \sigma_e^2 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \sigma_e^2 \end{bmatrix}$$

- Random-effect (G) structure

Linear Mixed Model
Residual structure
Random-effect (G) structure

We can think of our random effect specification in the same terms, and it's often called a G-structure (as opposed to the R-structure)

Linear Mixed Model

- Residual structure

$$\sigma_e^2 \mathbf{I} = \sigma_e^2 = \begin{bmatrix} \sigma_e^2 & 0 & 0 & \dots & 0 \\ 0 & \sigma_e^2 & 0 & \dots & 0 \\ 0 & 0 & \sigma_e^2 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \sigma_e^2 \end{bmatrix}$$

- Random-effect (G) structure

$$\sigma_u^2 \mathbf{Z}\mathbf{Z}^\top$$

We obtain it using this matrix equation. Don't panic. In simple models like we'll be fitting today what this term in blue does

Linear Mixed Model

- Residual structure

$$\sigma_e^2 \mathbf{I} = \sigma_e^2 = \begin{bmatrix} \sigma_e^2 & 0 & 0 & \dots & 0 \\ 0 & \sigma_e^2 & 0 & \dots & 0 \\ 0 & 0 & \sigma_e^2 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \sigma_e^2 \end{bmatrix}$$

- Random-effect (G) structure

$$\sigma_u^2 \mathbf{Z}\mathbf{Z}^\top = \sigma_u^2 \begin{bmatrix} 1 & 1 & 0 & \dots & 0 \\ 1 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 1 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 1 & 0 & 1 \end{bmatrix} =$$

Linear Mixed Model
Residual structure
Random-effect (G) structure

Is it puts a one along the diagonal and a one in any off-diagonal where two observations share the same level of the random effect. So for example, if you think about the 44 photos we scored we might add person as a random effect into the model because each person was photographed twice. This matrix now indicates that photo 1 and 2 are of the same person, and photo 3 and photo 44 are of the same person (in the actual data set photo's 3 and 44 are not of the same person).

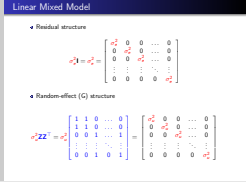
Linear Mixed Model

- Residual structure

$$\sigma_e^2 \mathbf{I} = \sigma_e^2 = \begin{bmatrix} \sigma_e^2 & 0 & 0 & \dots & 0 \\ 0 & \sigma_e^2 & 0 & \dots & 0 \\ 0 & 0 & \sigma_e^2 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \sigma_e^2 \end{bmatrix}$$

- Random-effect (G) structure

$$\sigma_u^2 \mathbf{Z}\mathbf{Z}^\top = \sigma_u^2 \begin{bmatrix} 1 & 1 & 0 & \dots & 0 \\ 1 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 1 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \sigma_u^2 & 0 & 0 & \dots & 0 \\ 0 & \sigma_u^2 & 0 & \dots & 0 \\ 0 & 0 & \sigma_u^2 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \sigma_u^2 \end{bmatrix}$$



The random effect variance along the diagonal represents the fact that there is going to be noise in the scores across photo's depending on which person we have photographed.

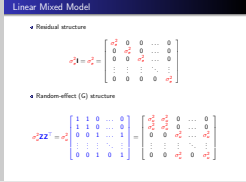
Linear Mixed Model

- Residual structure

$$\sigma_e^2 \mathbf{I} = \sigma_e^2 = \begin{bmatrix} \sigma_e^2 & 0 & 0 & \dots & 0 \\ 0 & \sigma_e^2 & 0 & \dots & 0 \\ 0 & 0 & \sigma_e^2 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \sigma_e^2 \end{bmatrix}$$

- Random-effect (G) structure

$$\sigma_u^2 \mathbf{Z}\mathbf{Z}^\top = \sigma_u^2 \begin{bmatrix} 1 & 1 & 0 & \dots & 0 \\ 1 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 1 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \sigma_u^2 & \sigma_u^2 & 0 & \dots & 0 \\ \sigma_u^2 & \sigma_u^2 & 0 & \dots & 0 \\ 0 & 0 & \sigma_u^2 & \dots & \sigma_u^2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \sigma_u^2 & 0 & \sigma_u^2 \end{bmatrix}$$



The off-diagonal tells us that variation caused by which person has been photographed is perfectly correlated between these two observations, because it is the same person that has been measured. I say perfectly correlated because a correlation is defined as the covariance between two numbers divided by the product of their standard deviations and $\sigma_u^2/(\sigma_u\sigma_u)$=1. The matrix $\mathbf{Z}\mathbf{Z}^\top$ is a correlation matrix. This doesn't imply that the actual observations will be perfectly correlated

Linear Mixed Model

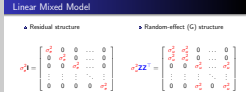
- Residual structure

$$\sigma_e^2 \mathbf{I} = \begin{bmatrix} \sigma_e^2 & 0 & 0 & \dots & 0 \\ 0 & \sigma_e^2 & 0 & \dots & 0 \\ 0 & 0 & \sigma_e^2 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \sigma_e^2 \end{bmatrix}$$

- Random-effect (G) structure

$$\sigma_u^2 \mathbf{Z}\mathbf{Z}^\top = \begin{bmatrix} \sigma_u^2 & \sigma_u^2 & 0 & \dots & 0 \\ \sigma_u^2 & \sigma_u^2 & 0 & \dots & 0 \\ 0 & 0 & \sigma_u^2 & \dots & \sigma_u^2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \sigma_u^2 & 0 & \sigma_u^2 \end{bmatrix}$$

Linear Mixed Model



We can take our R-structure which describes the variances (and possibly the covariances) in our data due to residual effects, and we can take our G-structure which describes the variances and the covariances in our data due to the random effects, and we can add them together

Linear Mixed Model

- Residual structure

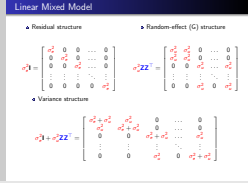
$$\sigma_e^2\mathbf{I} = \begin{bmatrix} \sigma_e^2 & 0 & 0 & \dots & 0 \\ 0 & \sigma_e^2 & 0 & \dots & 0 \\ 0 & 0 & \sigma_e^2 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \sigma_e^2 \end{bmatrix}$$

- Random-effect (G) structure

$$\sigma_u^2\mathbf{Z}\mathbf{Z}^\top = \begin{bmatrix} \sigma_u^2 & \sigma_u^2 & 0 & \dots & 0 \\ \sigma_u^2 & \sigma_u^2 & 0 & \dots & 0 \\ 0 & 0 & \sigma_u^2 & \dots & \sigma_u^2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \sigma_u^2 & 0 & \sigma_u^2 \end{bmatrix}$$

- Variance structure

$$\sigma_e^2\mathbf{I} + \sigma_u^2\mathbf{Z}\mathbf{Z}^\top = \begin{bmatrix} \sigma_e^2+\sigma_u^2 & \sigma_u^2 & 0 & \dots & 0 \\ \sigma_u^2 & \sigma_e^2+\sigma_u^2 & 0 & \dots & 0 \\ 0 & 0 & \sigma_e^2+\sigma_u^2 & \dots & \sigma_u^2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \sigma_u^2 & 0 & \sigma_e^2+\sigma_u^2 \end{bmatrix}$$



to get the variances and the covariances in our data after accounting for the fixed effects. Because the residuals are independent across observations they do not contribute to the covariances, but they do increase the variance, and so the expected correlation between the scores for photo 1 and photo 2 of the same person is $\sigma_u^2/(\sigma_u^2+\sigma_e^2)$. This might be expressed as the proportion of variance explained by that set of random effects, in this case the identities of the people photographed. However, the denominator here is the total variance after taking into account the fixed effects. If you are a good biologist and have gone out and meausred some key variables that determine the response, both σ_e^2 and σ_u^2 may decline depending on whether those variables vary within and/or between observations from photos of the same people.

Linear Mixed Model

```
> traffic_m6 <- lmer(y ~ limit + year + day + (1 | day), data = Traffic)
```

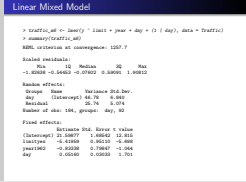

Linear Mixed Model

> traffic_m6 <- lmer(y ~ limit + year + day + (1 | day), data = Traffic)

Lets forget about bird nests and photos for a bit and get back to Swedish road accidents. Rather than use `lm` we're going to use the function `lmer` which fits linear mixed effect models (i.e. models were the distribution is assumed Gaussian, and there are both fixed and random effects - hence 'mixed' effects). The model syntax is similar to the one we used when fitting the model using `lm` but we've added the term `(1|day)` which fits day effect as random, we'll come back to why we have this 1 and a pipe later.

Linear Mixed Model

```
> traffic_m6 <- lmer(y ~ limit + year + day + (1 | day), data = Traffic)
> summary(traffic_m6)
REML criterion at convergence: 1257.7

Scaled residuals:
     Min       1Q   Median       3Q      Max
-1.82638 -0.54453 -0.07602  0.59091  1.90812

Random effects:
 Groups   Name        Variance Std.Dev.
 day      (Intercept) 46.78    6.840
 Residual             25.74    5.074
Number of obs: 184, groups:  day, 92

Fixed effects:
            Estimate Std. Error t value
(Intercept) 21.59877    1.68542  12.815
limityes    -5.41959    0.95110  -5.698
year1962    -0.83338    0.79847  -1.044
day          0.05160    0.03033   1.701
```

and we can get the summary of the model. Again we have the four fixed effects down here: the random day effects aren't shown (you can get them of course) but looking at a table of 92 day effects is not going to be very instructive. It is interesting to know that there are less accidents when a speed limit is in place, but you are probably not very interested in knowing, 'hey day 46 has more accidents than day 72!'. But you might be more interested in knowing generally whether days of the year do differ in the number of road accidents. And we can get a feel for this by looking at the estimated variance in the day effects: so about 46.8 compared to a residual variance of 25.7: so about 2/3rds of the variance in the number of accidents (after accounting for speed limit, year and a continuous trend across each year) can be explained by the day of the year.

Linear Mixed Model

```
> coef(summary(traffic_m6))
               Estimate Std. Error   t value
(Intercept) 21.59876943 1.68541510 12.815104
limityes    -5.41959096 0.95109962 -5.698237
year1962    -0.83338091 0.79847279 -1.043719
day          0.05159528 0.03033237  1.700998
```


Linear Mixed Model

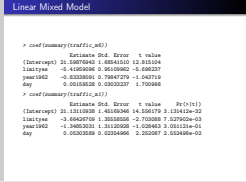
> coef(summary(traffic_m6))
Estimate Std. Error t value
(Intercept) 21.59876943 1.68541510 12.815104
limityes -5.41959096 0.95109962 -5.698237
year1962 -0.83338091 0.79847279 -1.043719
day 0.05159528 0.03033237 1.700998

We'll come back to the variance components, but for now lets take a look at the fixed effects. Lets start by comparing the coefficient table with what

```
> coef(summary(traffic_m6))
               Estimate Std. Error   t value
(Intercept) 21.59876943 1.68541510 12.815104
limityes    -5.41959096 0.95109962 -5.698237
year1962    -0.83338091 0.79847279 -1.043719
day          0.05159528 0.03033237  1.700998
> coef(summary(traffic_m1))
               Estimate Std. Error   t value     Pr(>|t|)
(Intercept) 21.13110938 1.45169346 14.556179 3.131412e-32
limityes    -3.66426709 1.35558556 -2.703088 7.527902e-03
year1962    -1.34853031 1.31120928 -1.028463 3.051121e-01
day          0.05303589 0.02354966  2.252087 2.552498e-02
```

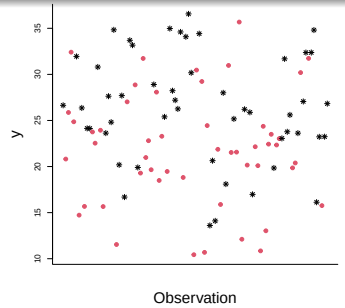

Random Effects (I)

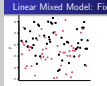
Linear Mixed Model



we had when when we didn't fit random day effects. The estimates have moved around a little: having a speed limit is estimated to reduce the number of accidents by 5.4 rather than 3.7, the difference between years seems to be less extreme than what we had originally thought, and the continuous trend with day is roughly comparable. This is to be expected; the estimates should differ (except in balanced cases) but they are as likely to go up as down. If we look at the standard errors we also see differences: the standard errors for the effect of speed limit has gone down by quite a bit (by 30%) as has the standard error for the year effect. However, the standard error for the continuous time trend is actually 1.29 times higher than before. Unlike the estimates themselves we do expect the standard errors to move in particular directions depending on how our predictor variables vary within and between levels of the random effect predictor (days in this case).

Linear Mixed Model: Fixed Effect Standard Errors

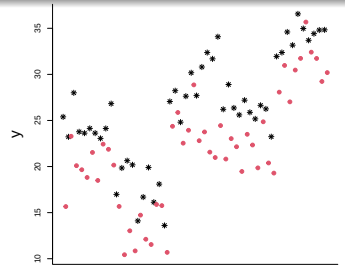




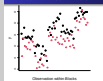
OK- so why might this be the case? Here are some data I've collected. The values are on the y axis, and the x axis doesn't really mean anything it's just the order in which I data-entered the values. Do you think that the black and red points have different means? Very hard to say - not obviously so.

When I was running this experiment I couldn't do everything on the same day so I had to do it over 5 days - in 5 blocks - using 20 individuals each day. However, what I did was that I took those 20 individuals and divided them randomly into two groups of ten and assigned one to the red treatment and one to the black treatment.

Linear Mixed Model: Fixed Effect Standard Errors

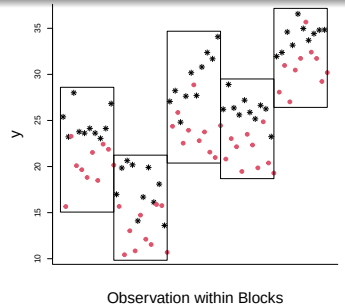


Linear Mixed Model: Fixed Effect Standard Errors

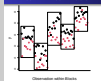


If I reorder the data so I put data points from the same day (block) next to each other

Linear Mixed Model: Fixed Effect Standard Errors

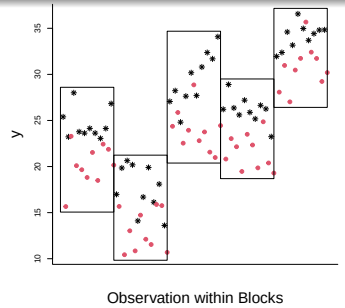


Linear Mixed Model: Fixed Effect Standard Errors



Now what do you think? Pretty convincing - if we look within days the red data points are consistently below the black ones and we can be fairly confident that there was an effect. The pattern was not obvious before because the between day variance was so large that it obscured the treatment differences. However, by applying our two treatments within days we did a very clever thing. It allows us to compare red and black points *within* a day, where the standard error on the estimate of the difference would only depend on the within day variance, not the total variance. If the within-day variance is small compared to the total variance this allows us to get much more precise estimates of the difference.

Linear Mixed Model: Fixed Effect Standard Errors



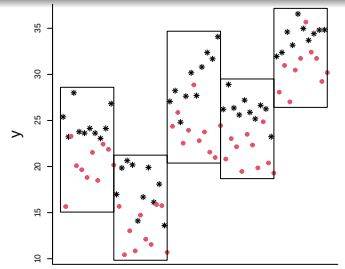
- Applying *both* treatments within blocks, standard error of the difference goes down.

Linear Mixed Model: Fixed Effect Standard Errors

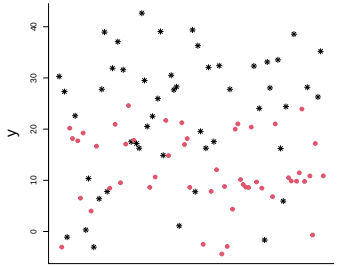
Applying *both* treatments within blocks, standard error of the difference goes down.

This is called blocking; if it's possible to do it then do it because you will get more precise estimates. I think most experimentalists are aware of this. But some times, particularly if you are collecting data in the field, it's not always possible to do this.

Linear Mixed Model: Fixed Effect Standard Errors



Observation within Blocks

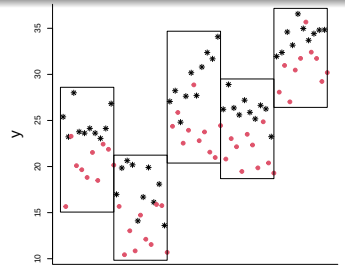


Observations

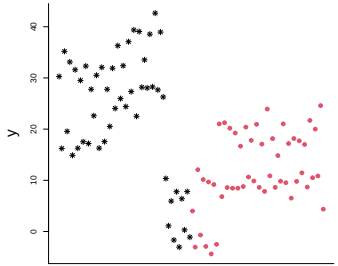
- Applying *both* treatments within blocks, standard error of the difference goes down.

The next year I went out and did another experiment - what do you think? Looks good - the red points seem to be quite substantially lower than the black points.

Linear Mixed Model: Fixed Effect Standard Errors

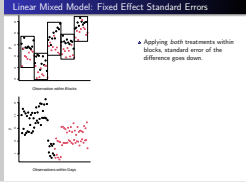


Observation within Blocks



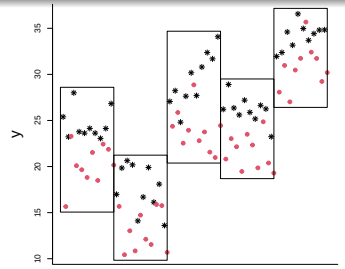
Observations within Days

- Applying *both* treatments within blocks, standard error of the difference goes down.

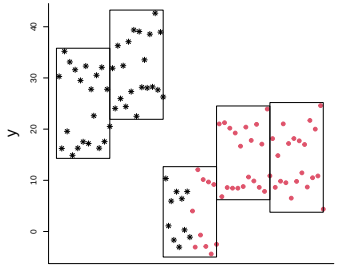


If we order our observations again by day

Linear Mixed Model: Fixed Effect Standard Errors



Observation within Blocks

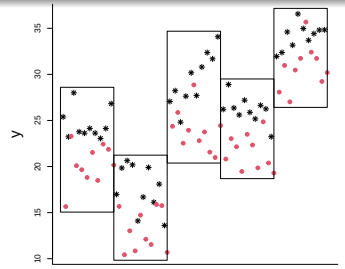


Observations within Days

- Applying *both* treatments within blocks, standard error of the difference goes down.

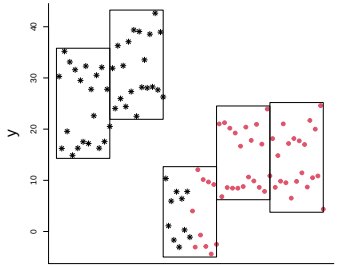
what do you think? This year, I couldn't be arsed to do both treatments each day. For the first two days I applied the black treatment, day 3 I did half and half (randomised of course!) and the last 2 days I applied the red treatment. I'm not so convinced now. Perhaps the effect was driven by differences between days.

Linear Mixed Model: Fixed Effect Standard Errors



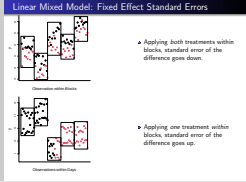
Observation within Blocks

- Applying *both* treatments within blocks, standard error of the difference goes down.



Observations within Days

- Applying *one* treatment *within* blocks, standard error of the difference goes up.



If our treatments predominantly vary between blocks rather than with-in blocks the standard error goes up. It's much harder to tell if there is a difference between the two treatments because that difference is partly confounded with between-block effects. Under this scenario people would say we have pseudoreplication and if we didn't deal with the block effects (either by averaging observations within blocks or estimating the block effects) our inferences about the treatment effects would be anti-conservative; we might declare treatment effects when in fact they are block effects. Conceptually, this is no different from the effects of confounding we covered when considering the effect of age and time in academia on grumpy scores.

Linear Mixed Model: Fixed Effect Standard Errors

```
> coef(summary(traffic_m6))
```

	Estimate	Std. Error	t value
(Intercept)	21.59876943	1.68541510	12.815104
limityes	-5.41959096	0.95109962	-5.698237
year1962	-0.83338091	0.79847279	-1.043719
day	0.05159528	0.03033237	1.700998

```
> coef(summary(traffic_m1))
```

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	21.13110938	1.45169346	14.556179	3.131412e-32
limityes	-3.66426709	1.35558556	-2.703088	7.527902e-03
year1962	-1.34853031	1.31120928	-1.028463	3.051121e-01
day	0.05303589	0.02354966	2.252087	2.552498e-02

Linear Mixed Model: Fixed Effect Standard Errors

> coef(summary(traffic_m6))

	Estimate	Std. Error	t value
(Intercept)	21.59876943	1.68541510	12.815104
limityes	-5.41959096	0.95109962	-5.698237
year1962	-0.83338091	0.79847279	-1.043719
day	0.05159528	0.03033237	1.700998

> coef(summary(traffic_m1))

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	21.13110938	1.45169346	14.556179	3.131412e-32
limityes	-3.66426709	1.35558556	-2.703088	7.527902e-03
year1962	-1.34853031	1.31120928	-1.028463	3.051121e-01
day	0.05303589	0.02354966	2.252087	2.552498e-02

So now we can understand why the standard errors have changed in the direction they have.

Linear Mixed Model: Fixed Effect Standard Errors

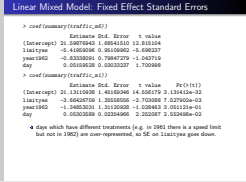
```
> coef(summary(traffic_m6))
```

	Estimate	Std. Error	t value
(Intercept)	21.59876943	1.68541510	12.815104
limityes	-5.41959096	0.95109962	-5.698237
year1962	-0.83338091	0.79847279	-1.043719
day	0.05159528	0.03033237	1.700998

```
> coef(summary(traffic_m1))
```

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	21.13110938	1.45169346	14.556179	3.131412e-32
limityes	-3.66426709	1.35558556	-2.703088	7.527902e-03
year1962	-1.34853031	1.31120928	-1.028463	3.051121e-01
day	0.05303589	0.02354966	2.252087	2.552498e-02

- days which have different treatments (e.g. in 1961 there is a speed limit but not in 1962) are over-represented, so SE on `limityes` goes down.



If speed limits were randomly assigned then we would expect the same day in two different years to differ in whether they had a speed limit or not 47% of the time[1]. In fact 64% of days differed in whether they had a speed limit or not, and so the treatment varies more with-in the block (day) than between. Because of this, and because the between-day effects are quite strong, the standard error has gone down. I'm not sure if this was design or by luck.

[1] In the data frame there was no speed limit on 62.5% of dates, and a speed limit on 37.5% of dates. If the researchers had picked at random whether a speed limit would be applied on any particular date for any pair of dates (corresponding to the same day but in different years) we would expect them to have different treatments with probability 2*0.625*0.375=0.47.

Linear Mixed Model: Fixed Effect Standard Errors

```
> coef(summary(traffic_m6))
               Estimate Std. Error   t value
(Intercept) 21.59876943 1.68541510 12.815104
limityes    -5.41959096 0.95109962 -5.698237
year1962    -0.83338091 0.79847279 -1.043719
day          0.05159528 0.03033237  1.700998
> coef(summary(traffic_m1))
               Estimate Std. Error   t value     Pr(>|t|)
(Intercept) 21.13110938 1.45169346 14.556179 3.131412e-32
limityes    -3.66426709 1.35558556 -2.703088 7.527902e-03
year1962    -1.34853031 1.31120928 -1.028463 3.051121e-01
day          0.05303589 0.02354966  2.252087 2.552498e-02
```

- days which have different treatments (e.g. in 1961 there is a speed limit but not in 1962) are over-represented, so SE on `limityes` goes down.
- every day has both years represented, so SE on `year1962` goes down.

Linear Mixed Model: Fixed Effect Standard Errors

All of the variation in year is within-block because every day has both years represented, and hence the standard error on the year effect goes down.

Linear Mixed Model: Fixed Effect Standard Errors

```
> coef(summary(traffic_m6))
               Estimate Std. Error   t value
(Intercept) 21.59876943 1.68541510 12.815104
limityes    -5.41959096 0.95109962 -5.698237
year1962    -0.83338091 0.79847279 -1.043719
day          0.05159528 0.03033237  1.700998
> coef(summary(traffic_m1))
               Estimate Std. Error   t value     Pr(>|t|)
(Intercept) 21.13110938 1.45169346 14.556179 3.131412e-32
limityes    -3.66426709 1.35558556 -2.703088 7.527902e-03
year1962    -1.34853031 1.31120928 -1.028463 3.051121e-01
day          0.05303589 0.02354966  2.252087 2.552498e-02
```

- days which have different treatments (e.g. in 1961 there is a speed limit but not in 1962) are over-represented, so SE on `limityes` goes down.
- every day has both years represented, so SE on `year1962` goes down.
- day as a continuous variable has no within day variance only between day variance (by definition!), so SE on `day` goes up.

Linear Mixed Model: Fixed Effect Standard Errors

> coef(summary(traffic_m6))
Estimate Std. Error t value
(Intercept) 21.59876943 1.68541510 12.815104
limityes -5.41959096 0.95109962 -5.698237
year1962 -0.83338091 0.79847279 -1.043719
day 0.05159528 0.03033237 1.700998
> coef(summary(traffic_m1))
Estimate Std. Error t value Pr(>|t|)
(Intercept) 21.13110938 1.45169346 14.556179 3.131412e-32
limityes -3.66426709 1.35558556 -2.703088 7.527902e-03
year1962 -1.34853031 1.31120928 -1.028463 3.051121e-01
day 0.05303589 0.02354966 2.252087 2.552498e-02

- days which have different treatments (e.g. in 1961 there is a speed limit but not in 1962) are over-represented, so SE on limityes goes down.
- every day has both years represented, so SE on year1962 goes down.
- day as a continuous variable has no within day variance only between day variance [by definition!], so SE on day goes up.

The opposite is true for day fitted as continuous variable; by definition the within-block variance is zero (both observations are made on the same day) and so the standard error goes up. So we can understand why the standard errors have changed but why are no p-values reported?

Linear Mixed Model: Fixed Effect Hypothesis testing

```
> coef(summary(traffic_m6))
               Estimate Std. Error   t value
(Intercept) 21.59876943 1.68541510 12.815104
limityes    -5.41959096 0.95109962 -5.698237
year1962    -0.83338091 0.79847279 -1.043719
day          0.05159528 0.03033237  1.700998
```


Linear Mixed Model: Fixed Effect Hypothesis testing

> coef(summary(traffic_m6))
Estimate Std. Error t value
(Intercept) 21.59876943 1.68541510 12.815104
limityes -5.41959096 0.95109962 -5.698237
year1962 -0.83338091 0.79847279 -1.043719
day 0.05159528 0.03033237 1.700998

A t-value is reported, as it is in the liner model without random effects, so why doesn't the summary report p-values from a t-test if the t-value is available. Recall that in a standard linear model the z-test assumes that the sampling distribution of a fixed effect is normal. This assumption is valid if our estimate of the residual variance is close to its true value. The t-test relaxes this assumption of the z-test and accounts for uncertainty in the residual variance. When this assumption is relaxed the sampling distribution is t-distributed.

When we use random effect models we have two variances rather than one. If our estimates of both variances are close to their true values then the sampling distribution of the fixed effects will remain normal and we can apply the z-test. If there is uncertainty in the variance components then the sampling distribution won't be normal, but (except in certain cases such as a paired design) it won't be a t-distribution either. In fact the sampling distribution is not a known distribution. A t-value (an estimate divided by its standard error) can still be reported of course, but given it doesn't come from a t-distribution there seems to be little reason to apply a t-test to it. Moreover, we don't know what the degrees of freedom should be.

So why give it? Well, the sampling distribution of fixed effects in a mixed effect model kind of 'look' like they're t-distributed, and as we saw earlier the t-distribution is quite insensitive to the degrees of freedom as long it is above 20 or so.

Linear Mixed Model: Fixed Effect Hypothesis testing

```
> coef(summary(traffic_m6))
               Estimate Std. Error   t value
(Intercept) 21.59876943 1.68541510 12.815104
limityes    -5.41959096 0.95109962 -5.698237
year1962    -0.83338091 0.79847279 -1.043719
day          0.05159528 0.03033237  1.700998
```

z-test

```
> 2 * (1 - pnorm(abs(coef(summary(traffic_m6))["day", "t value"])))
[1] 0.08894342
```


Linear Mixed Model: Fixed Effect Hypothesis testing

> coef(summary(traffic_m6))
Estimate Std. Error t value
(Intercept) 21.59876943 1.68541510 12.815104
limityes -5.41959096 0.95109962 -5.698237
year1962 -0.83338091 0.79847279 -1.043719
day 0.05159528 0.03033237 1.700998

z-test

> 2 * (1 - pnorm(abs(coef(summary(traffic_m6))["day", "t value"])))
[1] 0.08894342

So one possibility is to just use a z-test, and we can now see that the continuous year effect is now non-significant, in contrast (just) to the standard linear model.

Linear Mixed Model: Fixed Effect Hypothesis testing

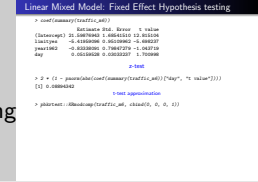
```
> coef(summary(traffic_m6))
               Estimate Std. Error   t value
(Intercept) 21.59876943 1.68541510 12.815104
limityes    -5.41959096 0.95109962 -5.698237
year1962    -0.83338091 0.79847279 -1.043719
day          0.05159528 0.03033237  1.700998
```

z-test

```
> 2 * (1 - pnorm(abs(coef(summary(traffic_m6))["day", "t value"])))
[1] 0.08894342
```

t-test approximation

```
> pbkrtest::KRmodcomp(traffic_m6, cbind(0, 0, 0, 1))
```

Another possible solution is to try and get a closer approximation to the sampling distribution using the Kenward-Roger approximation. This method can also be used to test whether multiple effects are zero as in an F-test type approach. Here we have a row vector equal in length to the number of fixed effects and each element is either a zero (we don't want to test the effect) or one (we do want to test the effect).

Linear Mixed Model: Fixed Effect Hypothesis testing

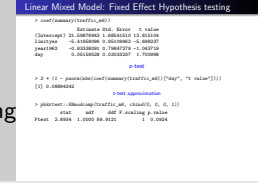
```
> coef(summary(traffic_m6))
               Estimate Std. Error   t value
(Intercept) 21.59876943 1.68541510 12.815104
limityes    -5.41959096 0.95109962 -5.698237
year1962    -0.83338091 0.79847279 -1.043719
day          0.05159528 0.03033237  1.700998
```

z-test

```
> 2 * (1 - pnorm(abs(coef(summary(traffic_m6))["day", "t value"])))
[1] 0.08894342
```

t-test approximation

```
> pbkrtest::KRmodcomp(traffic_m6, cbind(0, 0, 0, 1))
         stat     ndf     ddf F.scaling p.value
Ftest  2.8934  1.0000 89.9121         1  0.0924
```

As you can see, the p-value is very close to what we get using the z-test, which is what we expect unless our study is small or poorly designed.

You can also see that the denominator degrees of freedom is not an integer. In a standard linear model the denominator degrees of freedom in the F-test was the number of data points minus the number of 'fixed' effects estimated, so 184-4=180 in this case. The Kenward-Roger approximation can be thought of as a way of estimating the number of effective observations there are. Imagine that the day variance was massive compared to the residual variation meaning that essentially the same day on different years are identical. Imagine also that the thing your are testing does not vary within days (such as day as a continuous covariate). If this were the case you only have 1 effective data point rather 2, and so in total you only have 92 data points. Alternatively, if the day variance was exactly zero then the same day on different years are essentially independent and so you would have 2*92=184 data points in total. The reality will lie somewhere between these extremes, and depends on the magnitude of the between day variance relative to the residual variance. In this example the between day variance is large and so the Kenward-Roger approximation suggests there are few effective observations. The effective denominator degrees of freedom (89.9) plus the number of coefficients (4) gives an effective sample size of around 93.9, not much more than the number of days (92) and considerably less than the number of observations. For effects that vary within days the effective sample size can be more. For example, for the speed-limit effect the effective sample size is 114.4.

Linear Mixed Model: Fixed Effect Hypothesis testing

```
> coef(summary(traffic_m6))
               Estimate Std. Error   t value
(Intercept) 21.59876943 1.68541510 12.815104
limityes    -5.41959096 0.95109962 -5.698237
year1962    -0.83338091 0.79847279 -1.043719
day          0.05159528 0.03033237  1.700998
```

z-test

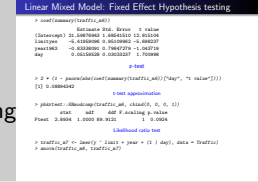
```
> 2 * (1 - pnorm(abs(coef(summary(traffic_m6))["day", "t value"])))
[1] 0.08894342
```

t-test approximation

```
> pbkrtest::KRmodcomp(traffic_m6, cbind(0, 0, 0, 1))
         stat     ndf     ddf F.scaling p.value
Ftest  2.8934  1.0000 89.9121         1  0.0924
```

Likelihood ratio test

```
> traffic_m7 <- lmer(y ~ limit + year + (1 | day), data = Traffic)
> anova(traffic_m6, traffic_m7)
```

We could also perform a likelihood ratio test where we compare the full model with a simpler model where the year effect is dropped. The function anova fits likelihood ratio tests by default when the model has been fitted by lmer[1].

[1] There are actually two likelihoods, which I'll come on to later; the standard likelihood, and something called a restricted likelihood. anova actually refits the models by maximising the standard likelihood (rather than the restricted likelihood, which is lmer's default) prior to comparing the models.

Linear Mixed Model: Fixed Effect Hypothesis testing

```
> coef(summary(traffic_m6))
               Estimate Std. Error   t value
(Intercept) 21.59876943 1.68541510 12.815104
limityes    -5.41959096 0.95109962 -5.698237
year1962    -0.83338091 0.79847279 -1.043719
day          0.05159528 0.03033237  1.700998
```

z-test

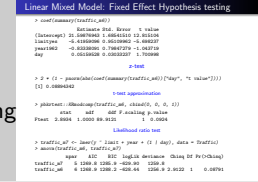
```
> 2 * (1 - pnorm(abs(coef(summary(traffic_m6))["day", "t value"])))
[1] 0.08894342
```

t-test approximation

```
> pbkrtest::KRmodcomp(traffic_m6, cbind(0, 0, 0, 1))
         stat     ndf     ddf F.scaling p.value
Ftest  2.8934  1.0000 89.9121         1  0.0924
```

Likelihood ratio test

```
> traffic_m7 <- lmer(y ~ limit + year + (1 | day), data = Traffic)
> anova(traffic_m6, traffic_m7)
          npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
traffic_m7    5 1269.8 1285.9 -629.90   1259.8
traffic_m6    6 1268.9 1288.2 -628.44   1256.9 2.9122  1    0.08791
```

and you can see again that the p-value is pretty close to those returned using other approaches, which is reassuring.

Deal with non-independence properly

Deal with non-independence properly

Science
AAAS

Recent natural selection causes adaptive evolution of an avian polygenic trait

Deal with non-independence properly

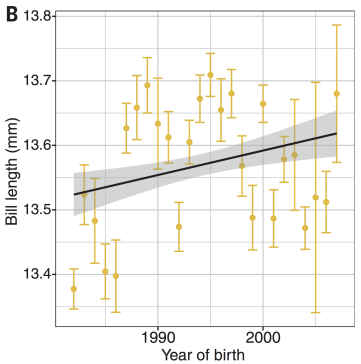
Science

Recent natural selection causes adaptive evolution of an avian polygenic trait

So hopefully I've convinced you that if you can apply treatments with-in blocks then this is a good way of increasing power. For example, if you wanted to know whether giving someone a present made them happy, you would be much better off having two observations per person (for example measuring everyone's happiness before the experiment, and then after). However, if the treatment or variable you are interested in varies a lot among blocks then you have to deal with it otherwise you may well claim an effect is there when the evidence is actually pretty weak.

Science
AAAS

Recent natural selection causes adaptive evolution of an avian polygenic trait



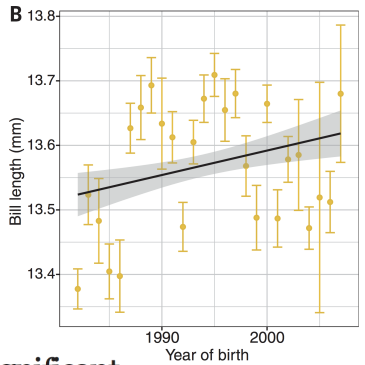
Deal with non-independence properly
Science
Recent natural selection causes adaptive evolution of an avian polygenic trait

This is a recent science paper claiming that bill length in British great tits has increased over the last 25 years, and this is an evolutionary response to using bird feeders.

Deal with non-independence properly

Science
AAAS

Recent natural selection causes adaptive evolution of an avian polygenic trait



we found that bill length has increased significantly over recent years (1982–2007; n = 2489 birds; estimate = 0.004 ± 0.001 mm per year; P = 0.0038;

Deal with non-independence properly

Science

Recent natural selection causes adaptive evolution of an avian polygenic trait

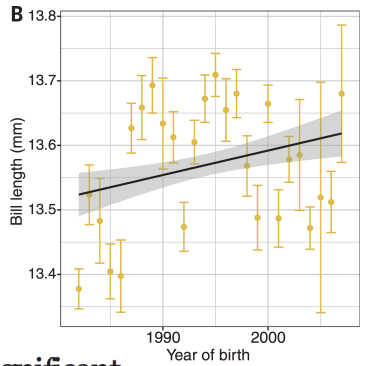
we found that bill length has increased significantly over recent years (1982–2007; n = 2489 birds; estimate = 0.004 ± 0.001 mm per year; P = 0.0038;

Each point on this graph is the mean bill-length in a particular year, and the whiskers are the standard errors. The dark line is their best estimate of the slope, and the shaded area the standard errors of the predicted mean.

Deal with non-independence properly

Science
AAAS

Recent natural selection causes adaptive evolution of an avian polygenic trait



we found that bill length has increased significantly over recent years (1982–2007; n = 2489 birds; estimate = 0.004 ± 0.001 mm per year; P = 0.0038;

is not due to stochastic variation among years (randomization test, P = 0.02)

Deal with non-independence properly

Science
AAAS

Recent natural selection causes adaptive evolution of an avian polygenic trait

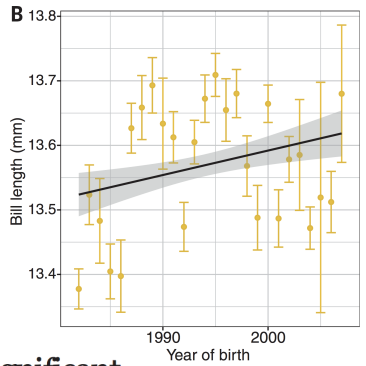
we found that bill length has increased significantly over recent years (1982–2007; n = 2489 birds; estimate = 0.004 ± 0.001 mm per year; P = 0.0038;

is not due to stochastic variation among years (randomization test, P = 0.02)

The estimated change is 0.004mm per year and so with a mean bill-length of around 13.5mm at the start this constitutes a change of about `(0.004/13.5)*100`=0.03% per year, or a `(25*0.004/13.5)*100`=0.74% change across the 25 years. A tiny change really ($<$ 1%) and considerably smaller than the between-year changes, but still highly significant.

Science
AAAS

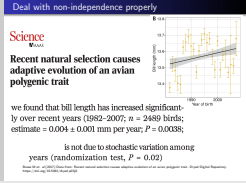
Recent natural selection causes adaptive evolution of an avian polygenic trait



we found that bill length has increased significantly over recent years (1982–2007; n = 2489 birds; estimate = 0.004 ± 0.001 mm per year; P = 0.0038;

is not due to stochastic variation among years (randomization test, P = 0.02)

Bosse M *et. al* (2017) Data from: Recent natural selection causes adaptive evolution of an avian polygenic trait. Dryad Digital Repository. https://doi.org/10.5061/dryad.p03j0



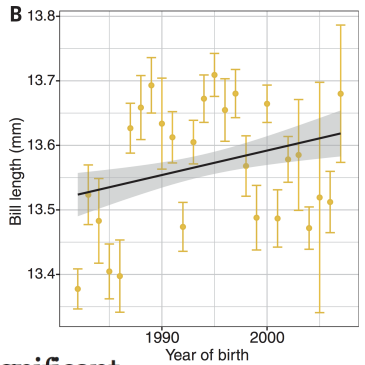
However, this slope and standard error are calculated without year effects being accounted for. However, after a randomisation test [1] they claim that the slope is still significantly different from zero.

[1] The randomisation test they perform is exactly how it should not be done and is very anti-conservative. They should have permuted the year labels rather than the year column. They have measured birds from 1982 to 2007 and what they should have done is jumbled up the year names so that, for example, 1982 becomes 1997. Then they should have replaced all instances of 1982 in the year column with the *same* value: 1997. What they actually did was replace all instances of 1982 with jumbled up values from the year column, such that birds originally measured in 1982 are assigned to a range of years. The way they did it they are testing whether there is any variation in bill-length across years, either random fluctuations *or* systematic changes with time consistent with an evolutionary response. If they had done it my way they would just be testing whether there are systematic changes with time.

Deal with non-independence properly

Science
AAAS

Recent natural selection causes adaptive evolution of an avian polygenic trait



we found that bill length has increased significantly over recent years (1982–2007; n = 2489 birds; estimate = 0.004 ± 0.001 mm per year; P = 0.0038;

is not due to stochastic variation among years (randomization test, P = 0.02)

Bosse M *et. al* (2017) Data from: Recent natural selection causes adaptive evolution of an avian polygenic trait. Dryad Digital Repository. https://doi.org/10.5061/dryad.p03j0

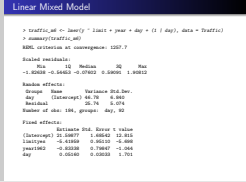
Deal with non-independence properly
Science
Recent natural selection causes adaptive evolution of an avian polygenic trait
we found that bill length has increased significantly over recent years (1982–2007; n = 2489 birds; estimate = 0.004 ± 0.001 mm per year; P = 0.0038;
is not due to stochastic variation among years (randomization test, P = 0.02)

The data are available on dryad, so have a go and analyse their data using what we've just learnt (the actual p-value from a mixed model using KRmodcomp is 0.27, close to the p-value obtained using the correct randomisation test).

```
m_bosse<-lmer(bl~yrbirth+(1|yrbirth), data=bosse)
pbkrtest::KRmodcomp(m_bosse, cbind(0,1))
```

Linear Mixed Model

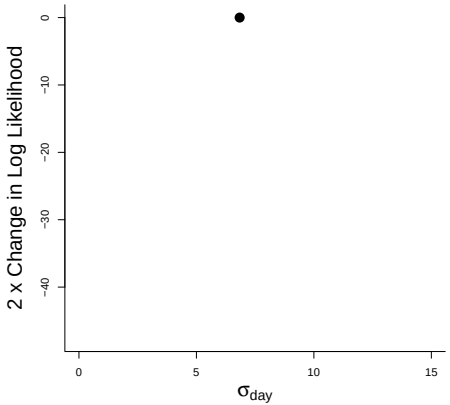
```
> traffic_m6 <- lmer(y ~ limit + year + day + (1 | day), data = Traffic)
> summary(traffic_m6)
REML criterion at convergence: 1257.7

Scaled residuals:
     Min       1Q   Median       3Q      Max
-1.82638 -0.54453 -0.07602  0.59091  1.90812

Random effects:
 Groups   Name        Variance Std.Dev.
 day      (Intercept) 46.78    6.840
 Residual             25.74    5.074
Number of obs: 184, groups:  day, 92

Fixed effects:
            Estimate Std. Error t value
(Intercept) 21.59877    1.68542  12.815
limityes    -5.41959    0.95110  -5.698
year1962    -0.83338    0.79847  -1.044
day          0.05160    0.03033   1.701
```

Now many people here will be primarily interested in the fixed effects, but I think also many people - although they might not be aware of it yet, will be interested in the variances. Our best estimate tells us about 2/3rds of the variance in the number of accidents (after accounting for speed limit, year and a continuous trend with in each year) can be explained by the day of the year. But you might wonder how precise our estimate of 2/3rds is. Is a value of half likely, or can we say it's definitely over half. One possibility is to assume that we have sufficient information about the variance parameter that the sampling distribution of the estimates are approximately normal. We could then use the resulting standard errors to inform us how precise the estimate are. At first glance you may think that this second column in the random effect summary are the standard errors - but no, they're simply the estimates of the variances square rooted (the estimates of the standard deviations). The problem is that you often need very large sample sizes before the sampling distribution of variance estimates start to look normal, particularly if the variance is close to zero, and so the normal-approximation is so approximate it's often not very useful. However, there are a number of other methods - which are bit more involved - for getting confidence intervals on these parameters.

Linear Mixed Model: Confidence Intervals

Profile Likelihood

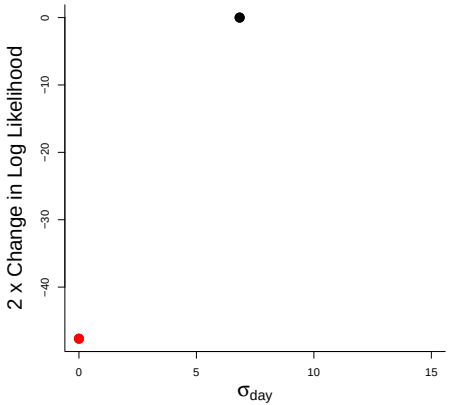


- Get the likelihood of the the model.

The method I would like to go through first involves a profile likelihood. On the x-axis we have our parameter of interest; in this case the standard deviation of the day effects (our variance square-rooted) and on the y axis we have the amount the log-likelihood (doubled) changes compared to our best fitting model. This point is our (restricted) maximum likelihood estimate.

Linear Mixed Model: Confidence Intervals

Profile Likelihood

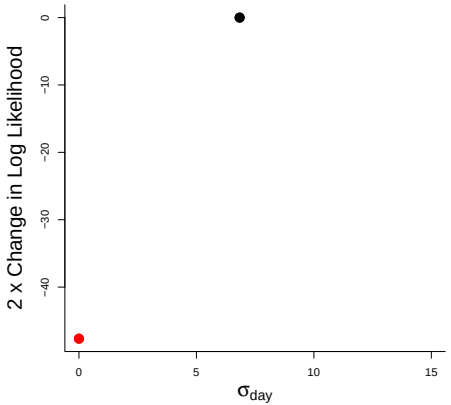


- Get the likelihood of the the model.

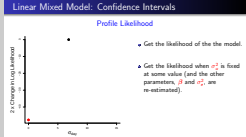
What we could then do is calculate by how much the log-likelihood (doubled) changes if we fix the day standard deviation at zero. In this case the log-likelihood has changed by almost 25.

Linear Mixed Model: Confidence Intervals

Profile Likelihood



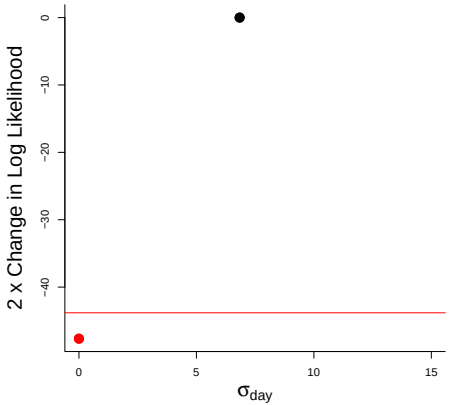
- Get the likelihood of the the model.
- Get the likelihood when σ_u^2 is fixed at some value (and the other parameters, β and σ_e^2, are re-estimated).



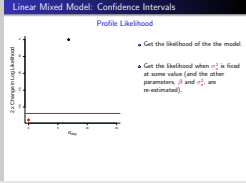
To obtain this likelihood, we've fixed the day standard deviation at zero, but allowed the remaining parameters to be re-estimated such that the likelihood is maximised conditional on the day standard deviation being zero.

Linear Mixed Model: Confidence Intervals

Profile Likelihood



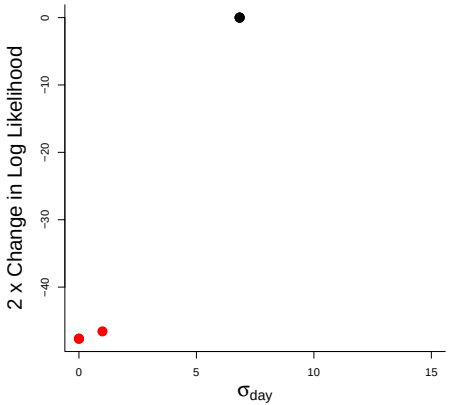
- Get the likelihood of the the model.
- Get the likelihood when σ_u^2 is fixed at some value (and the other parameters, β and σ_e^2, are re-estimated).



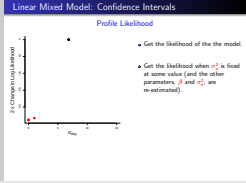
The y-axis is in units of *twice* the change in log-likelihood. We use this value because we have an approximation for how much this quantity should change when comparing a model with a parameter fixed versus estimated, had the data been generated by the model defined by the fixed parameter. This was the basis of the likelihood ratio test. This quantity should follow a chi-squared distribution with one degree of freedom. If we imagine our data really had been generated under a model where the day variance was zero, this red-line denotes the amount of change in *twice* the log-likelihoods we would observe between the two models (σ_{day} fixed at zero versus estimated) 5% of the time, just by chance. If our null model was that the day variance was zero, then a change in likelihood greater than this would be declared significant (at the 5% level), which is overwhelmingly supported in our case.

Linear Mixed Model: Confidence Intervals

Profile Likelihood



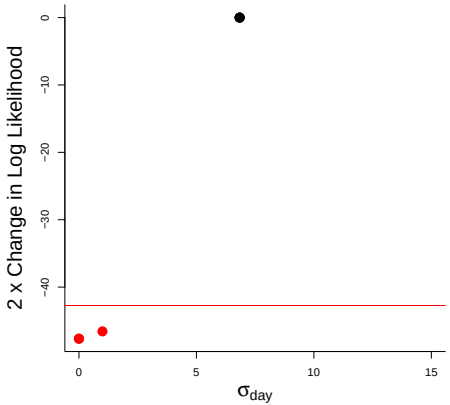
- Get the likelihood of the the model.
- Get the likelihood when σ_u^2 is fixed at some value (and the other parameters, β and σ_e^2, are re-estimated).



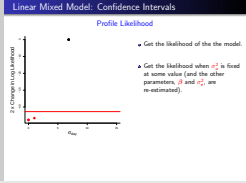
However, what we could do is rather than have a day standard deviation of zero as our null, we could fix the day standard deviation at something a bit higher.

Linear Mixed Model: Confidence Intervals

Profile Likelihood



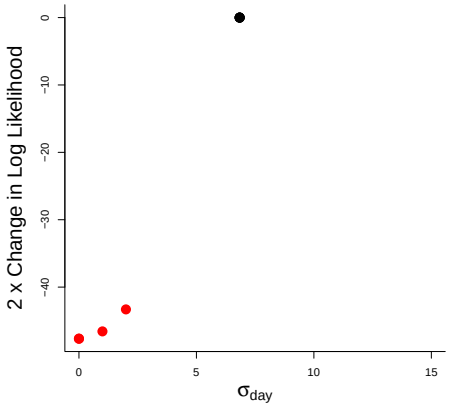
- Get the likelihood of the the model.
- Get the likelihood when σ_u^2 is fixed at some value (and the other parameters, β and σ_e^2, are re-estimated).



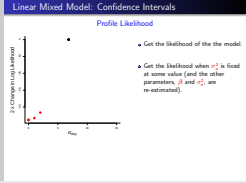
We could then redraw our line indicating the change in likelihood beyond which we would declare the models significantly different. Still the actual change we see between this new model and our best model exceeds this threshold.

Linear Mixed Model: Confidence Intervals

Profile Likelihood



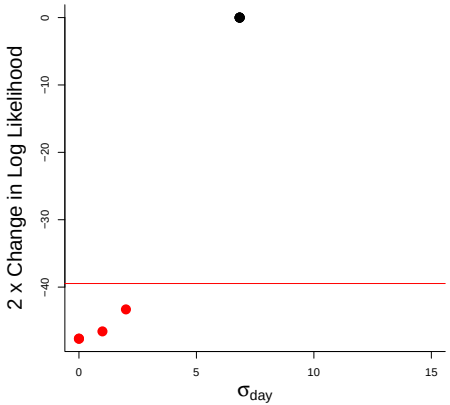
- Get the likelihood of the the model.
- Get the likelihood when σ_u^2 is fixed at some value (and the other parameters, β and σ_e^2, are re-estimated).



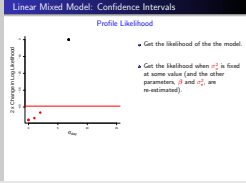
We could then increase the day standard deviation further

Linear Mixed Model: Confidence Intervals

Profile Likelihood



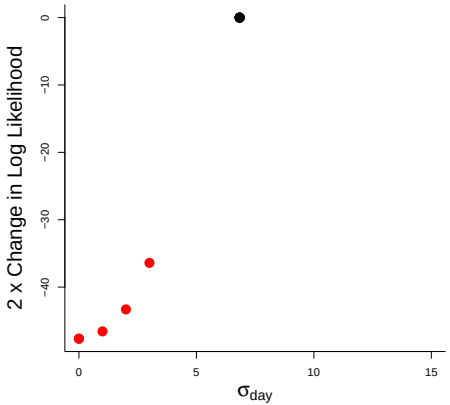
- Get the likelihood of the the model.
- Get the likelihood when σ_u^2 is fixed at some value (and the other parameters, β and σ_e^2, are re-estimated).



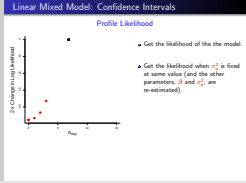
calculate the new threshold - and the actual change we see, still exceeds this threshold.

Linear Mixed Model: Confidence Intervals

Profile Likelihood



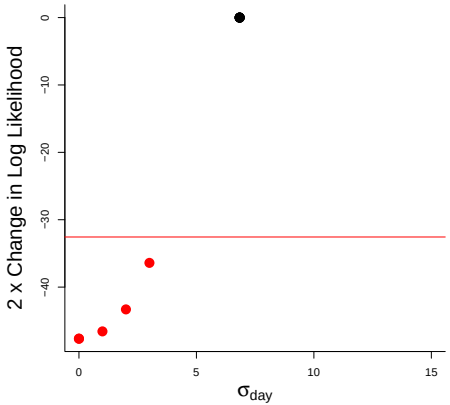
- Get the likelihood of the the model.
- Get the likelihood when σ_u^2 is fixed at some value (and the other parameters, β and σ_e^2, are re-estimated).



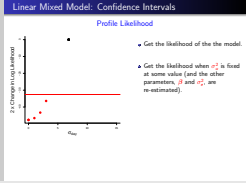
We could repeat this procedure

Linear Mixed Model: Confidence Intervals

Profile Likelihood



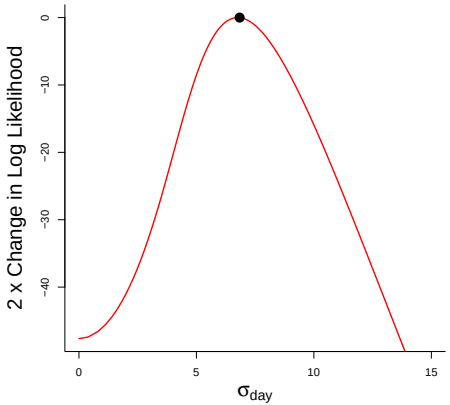
- Get the likelihood of the the model.
- Get the likelihood when σ_u^2 is fixed at some value (and the other parameters, β and σ_e^2, are re-estimated).



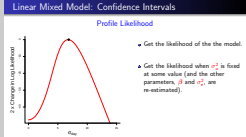
for a whole range of different values

Linear Mixed Model: Confidence Intervals

Profile Likelihood



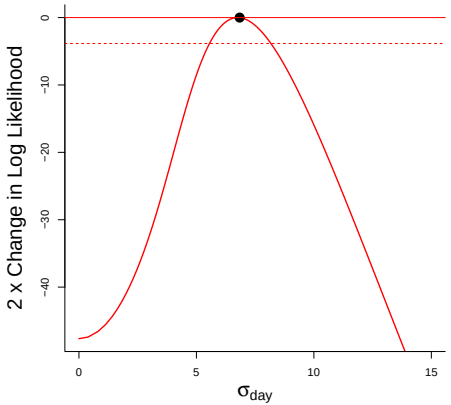
- Get the likelihood of the the model.
- Get the likelihood when σ_u^2 is fixed at some value (and the other parameters, β and σ_e^2, are re-estimated).



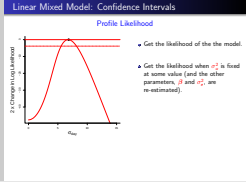
until we get this curve. This is the profile likelihood - how the likelihood changes as a function of a particular parameter, allowing the remaining parameters to take on the values that maximise this conditional likelihood.

Linear Mixed Model: Confidence Intervals

Profile Likelihood



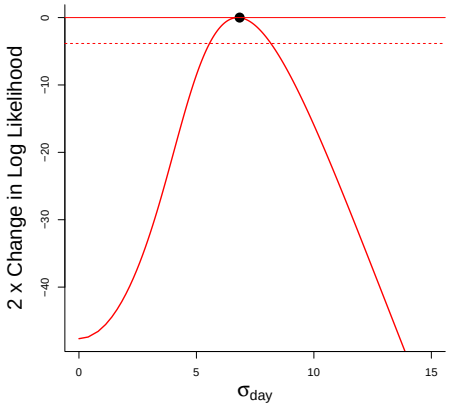
- Get the likelihood of the the model.
- Get the likelihood when σ_u^2 is fixed at some value (and the other parameters, β and σ_e^2, are re-estimated).



We could then draw a line below our peak

Linear Mixed Model: Confidence Intervals

Profile Likelihood



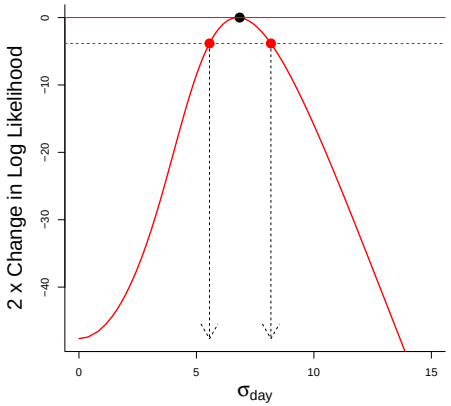
- Get the likelihood of the the model.
- Get the likelihood when σ_u^2 is fixed at some value (and the other parameters, β and σ_e^2, are re-estimated).
- The critical value is `qchisq(0.95,1)`=3.84 for the 95% confidence interval.

Linear Mixed Model: Confidence Intervals
Profile Likelihood
Get the likelihood of the the model.
Get the likelihood when σ_u^2 is fixed at some value (and the other parameters, β and σ_e^2, are re-estimated).
The critical value is qchisq(0.95,1)=3.84 for the 95% confidence interval.

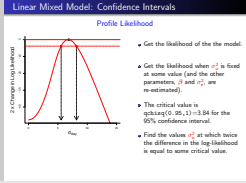
indicating the change in likelihood that would be deemed significant at the 5% level. At a 5% level we can find this value by asking below what value would 95% of values lie if they came from a chi-squared distribution with 1 degree of freedom. In this case it's 3.84.

Linear Mixed Model: Confidence Intervals

Profile Likelihood



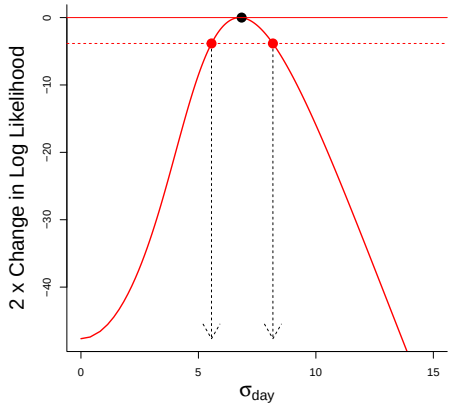
- Get the likelihood of the the model.
- Get the likelihood when σ_u^2 is fixed at some value (and the other parameters, β and σ_e^2, are re-estimated).
- The critical value is `qchisq(0.95,1)`=3.84 for the 95% confidence interval.
- Find the values σ_u^2 at which twice the difference in the log-likelihood is equal to some critical value.



We can then find the values of our parameter that satisfy this condition, and these would give us a 95% confidence interval. If you wanted a different confidence interval - an 80% confidence interval say - then you could change this critical value (`qchisq(0.8,1)`=1.64 in this case).

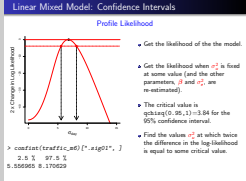
Linear Mixed Model: Confidence Intervals

Profile Likelihood



- Get the likelihood of the the model.
- Get the likelihood when σ_u^2 is fixed at some value (and the other parameters, β and σ_e^2, are re-estimated).
- The critical value is `qchisq(0.95,1)`=3.84 for the 95% confidence interval.
- Find the values σ_u^2 at which twice the difference in the log-likelihood is equal to some critical value.

```
> confint(traffic_m6)[".sig01", ]
    2.5 %    97.5 %
5.556965 8.170629
```

The function `confint` is a useful function for obtaining confidence intervals that can be applied to many models, and for models fit in `lmer` it defaults to the profile likelihood method. You simply pass it the model and it calculates profile-likelihood confidence intervals for all parameters. Here I've just presented the confidence interval for `".sig01"` which is the first standard deviation (`".sig"` is short for sigma) in the summary output (`day`).

Linear Mixed Model: Confidence Intervals

Parametric Bootstrap

- Simulate new data using the (restricted) maximum likelihood estimates ($\hat{\beta}$, $\hat{\sigma}_e^2$ and $\hat{\sigma}_u^2$).

Linear Mixed Model: Confidence Intervals

Parametric Bootstrap

- Simulate new data using the (restricted) maximum likelihood estimates ($\hat{\beta}$, $\hat{\sigma}_e^2$ and $\hat{\sigma}_u^2$).

another widely used technique is the parametric bootstrap - sounds fancy but it's very simple, and in fact you did it in the first practical. Basically: we take our parameter estimates, so beta and our two variances, and we simulate data according to these estimates.

Linear Mixed Model: Confidence Intervals

Parametric Bootstrap

- Simulate new data using the (restricted) maximum likelihood estimates ($\hat{\beta}$, $\hat{\sigma}_e^2$ and $\hat{\sigma}_u^2$).
- Refit model using new data to get the sampling distribution.

Linear Mixed Model: Confidence Intervals

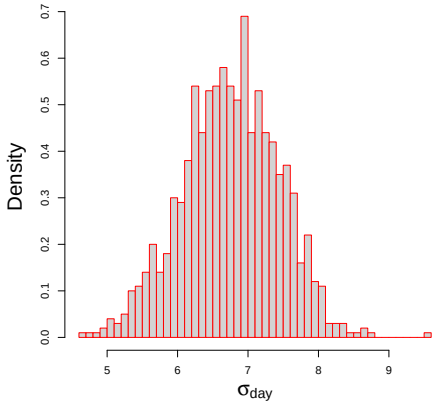
Parametric Bootstrap

- Simulate new data using the (restricted) maximum likelihood estimates ($\hat{\beta}$, $\hat{\sigma}_e^2$ and $\hat{\sigma}_u^2$).
- Refit model using new data to get the sampling distribution.

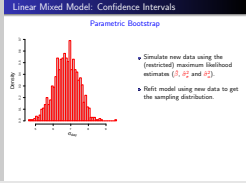
We then refit the model to these simulated data and re-estimate the parameters. The deviation of these new estimates around our actual estimates gives us a measure how much variation we expect just by chance.

Linear Mixed Model: Confidence Intervals

Parametric Bootstrap



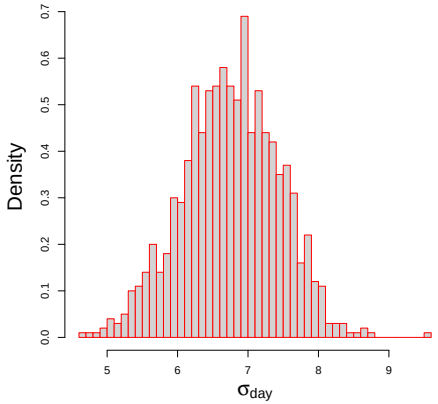
- Simulate new data using the (restricted) maximum likelihood estimates ($\hat{\beta}$, $\hat{\sigma}^2_e$ and $\hat{\sigma}^2_u$).
- Refit model using new data to get the sampling distribution.



This histogram was generated by using the function `simulate` on our model to generate a thousand new sets of accident data, and then refitting the model to each of these new sets and extracting the parameter estimates (for the day standard-deviation in this case).

Linear Mixed Model: Confidence Intervals

Parametric Bootstrap



- Simulate new data using the (restricted) maximum likelihood estimates ($\hat{\beta}$, $\hat{\sigma}^2_e$ and $\hat{\sigma}^2_u$).
- Refit model using new data to get the sampling distribution.
- Get the 2.5% and 97.5% quantiles

Linear Mixed Model: Confidence Intervals

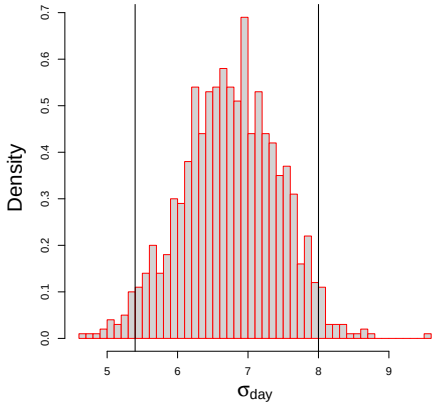
Parametric Bootstrap

- Simulate new data using the (restricted) maximum likelihood estimates ($\hat{\beta}$, $\hat{\sigma}^2_e$ and $\hat{\sigma}^2_u$).
- Refit model using new data to get the sampling distribution.
- Get the 2.5% and 97.5% quantiles

What we can then do is find the value beneath which 2.5% of the estimates fall, and the value above which 97.5% of the estimates fall,

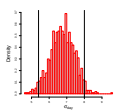
Linear Mixed Model: Confidence Intervals

Parametric Bootstrap



- Simulate new data using the (restricted) maximum likelihood estimates ($\hat{\beta}$, $\hat{\sigma}_e^2$ and $\hat{\sigma}_u^2$).
- Refit model using new data to get the sampling distribution.
- Get the 2.5% and 97.5% quantiles

Linear Mixed Model: Confidence Intervals
Parametric Bootstrap

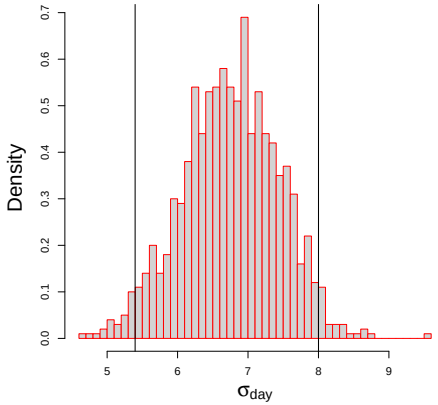


Simulate new data using the (restricted) maximum likelihood estimates ($\hat{\beta}$, $\hat{\sigma}_e^2$ and $\hat{\sigma}_u^2$).
Refit model using new data to get the sampling distribution.
Get the 2.5% and 97.5% quantiles

so these values, to obatin our 95% intervals.

Linear Mixed Model: Confidence Intervals

Parametric Bootstrap



- Simulate new data using the (restricted) maximum likelihood estimates ($\hat{\beta}$, $\hat{\sigma}^2_e$ and $\hat{\sigma}^2_u$).
- Refit model using new data to get the sampling distribution.
- Get the 2.5% and 97.5% quantiles

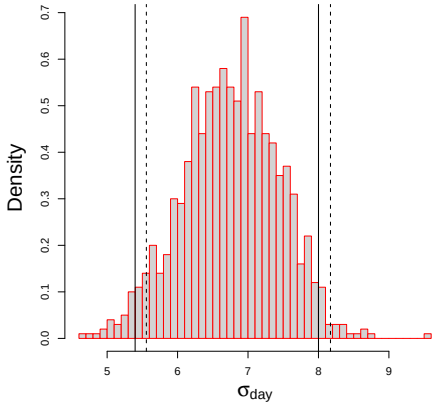
```
> confint(traffic_m6, method = "boot")[".sig01", ]
   2.5 %   97.5 %
5.624805 8.010794
```


Linear Mixed Model: Confidence Intervals
Parametric Bootstrap
Simulate new data using the (restricted) maximum likelihood estimates ($\hat{\beta}$, $\hat{\sigma}^2_e$ and $\hat{\sigma}^2_u$).
Refit model using new data to get the sampling distribution.
Get the 2.5% and 97.5% quantiles
> confint(traffic_m6, method = "boot")[".sig01",]
2.5 % 97.5 %
5.624805 8.010794

Again, you can use the function `confint` to do this by specifying `method="boot"`. And we can see that although our best estimate for the day variance was 46: values as low as 30 (5.62^2=31.64) and as high as 66 (8.01^2=64.17) are possible.

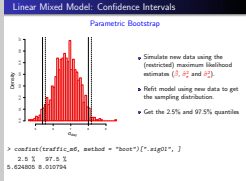
Linear Mixed Model: Confidence Intervals

Parametric Bootstrap



- Simulate new data using the (restricted) maximum likelihood estimates ($\hat{\beta}$, $\hat{\sigma}_e^2$ and $\hat{\sigma}_u^2$).
- Refit model using new data to get the sampling distribution.
- Get the 2.5% and 97.5% quantiles

```
> confint(traffic_m6, method = "boot")[".sig01", ]
   2.5 %   97.5 %
5.624805 8.010794
```

We can compare this to our confidence intervals using the profile likelihood, and we can see that they're quite close but a bit narrower. One nice property of the parametric bootsrap is that it's also possible to get confidence intervals on quantities that are functions of *multiple* parameters, and this is usually quite hard to do using a profile likelihood. For example, our point estimate for the proportion of variance explained by day effects is about 65%, but how would we get confidence intervals on this proportion?

Linear Mixed Model: Confidence Intervals

Parametric Bootstrap

```
> traffic_sim <- simulate(traffic_m6, 1000)
```


Linear Mixed Model: Confidence Intervals
Parametric Bootstrap
> traffic_sim <- simulate(traffic_m6, 1000)

As far as I know, there isn't a ready-made `confit` function for this problem, but we can easily roll one by hand. First, we can simulate a 1000 replicate data sets according to our model.

Linear Mixed Model: Confidence Intervals

Parametric Bootstrap

```
> traffic_sim <- simulate(traffic_m6, 1000)
> traffic_pb <- t(apply(traffic_sim, 2, function(x) {
+     coefv(refit(traffic_m6, x))
+ }))
```


Linear Mixed Model: Confidence Intervals
Parametric Bootstrap
> traffic_sim <- simulate(traffic_m6, 1000)
> traffic_pb <- t(apply(traffic_sim, 2, function(x) {
+ coefv(refit(traffic_m6, x))
+ }))

Then we can run this bit of code that requires a bit of unpacking. `trafic_sim` is a data frame with 1000 columns, each of which contains 184 simulated accident observations. The function `apply` takes each column of `trafic_sim` and passes it to the first argument of a `function` (this is because the second argument (`MARGIN`) is 2 - had it been a 1 it would have taken each row). In this case, this is my own function and it only takes one argument, `x`. This will be a replicate data set, and the function refits the model `traffic_m6` to the new data using the function `refit`, and then `coefv` extracts the estimates of the variance components from the refitted model. `coefv` is a function I have written:

```
coefv<-function(x, var=TRUE){
 if(!class(x)%in%c("lmerMod", "glmerMod")){
   stop("x should be a lmer or glmer model")
 }
 vrandom<-unlist(lapply(VarCorr(x), function(x){attr(x, "stddev")}))
 verror<-attr(VarCorr(x), "sc")
 names(verror)<-"Residual"
 v<-c(vrandom, verror)
 if(var){
   v<-v*v
 }
 return(v)
}
```

Linear Mixed Model: Confidence Intervals

Parametric Bootstrap

```
> traffic_sim <- simulate(traffic_m6, 1000)
> traffic_pb <- t(apply(traffic_sim, 2, function(x) {
+     coefv(refit(traffic_m6, x))
+ }))
> head(traffic_pb)
      day.(Intercept) Residual
sim_1        38.69853 30.42440
sim_2        43.12451 26.22378
sim_3        56.11658 22.15043
sim_4        52.20232 26.28133
sim_5        44.87838 23.92620
sim_6        44.30703 25.31061
```


Linear Mixed Model: Confidence Intervals

Parametric Bootstrap

> traffic_sim <- simulate(traffic_m6, 1000)
> traffic_pb <- t(apply(traffic_sim, 2, function(x) {
+ coefv(refit(traffic_m6, x))
+ }))
> head(traffic_pb)
day.(Intercept) Residual
sim_1 38.69853 30.42440
sim_2 43.12451 26.22378
sim_3 56.11658 22.15043
sim_4 52.20232 26.28133
sim_5 44.87838 23.92620
sim_6 44.30703 25.31061

`traffic_pb` then contains 1000 rows (because I have transposed the output of `apply` using the function `t`) with each row pertaining to a replicate data set, and the two numbers on that row being the day variance and the residual variance estimates made on that replicate data set.

Linear Mixed Model: Confidence Intervals

Parametric Bootstrap

```
> traffic_sim <- simulate(traffic_m6, 1000)
> traffic_pb <- t(apply(traffic_sim, 2, function(x) {
+     coefv(refit(traffic_m6, x))
+ }))
> head(traffic_pb)
```

	day.(Intercept)	Residual
sim_1	38.69853	30.42440
sim_2	43.12451	26.22378
sim_3	56.11658	22.15043
sim_4	52.20232	26.28133
sim_5	44.87838	23.92620
sim_6	44.30703	25.31061

```
> propday.pb <- traffic_pb[, 1]/rowSums(traffic_pb)
```


Linear Mixed Model: Confidence Intervals

Parametric Bootstrap

> traffic_sim <- simulate(traffic_m6, 1000)
> traffic_pb <- t(apply(traffic_sim, 2, function(x) {
+ coefv(refit(traffic_m6, x))
+ }))
> head(traffic_pb)

	day.(Intercept)	Residual
sim_1	38.69853	30.42440
sim_2	43.12451	26.22378
sim_3	56.11658	22.15043
sim_4	52.20232	26.28133
sim_5	44.87838	23.92620
sim_6	44.30703	25.31061

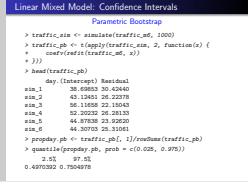
> propday.pb <- traffic_pb[, 1]/rowSums(traffic_pb)

What we can then do is take the first number and divide it by the sum of both numbers to get a proportion, and do this for each pair of numbers. This is stored in `propday.pb` which contains the 1000 estimated proportions.

Linear Mixed Model: Confidence Intervals

Parametric Bootstrap

```
> traffic_sim <- simulate(traffic_m6, 1000)
> traffic_pb <- t(apply(traffic_sim, 2, function(x) {
+     coefv(refit(traffic_m6, x))
+ }))
> head(traffic_pb)
      day.(Intercept) Residual
sim_1        38.69853 30.42440
sim_2        43.12451 26.22378
sim_3        56.11658 22.15043
sim_4        52.20232 26.28133
sim_5        44.87838 23.92620
sim_6        44.30703 25.31061
> propday.pb <- traffic_pb[, 1]/rowSums(traffic_pb)
> quantile(propday.pb, prob = c(0.025, 0.975))
     2.5%     97.5%
0.4970392 0.7504978
```

Finally, we can then get the 2.5% and 97.5% quantiles, and we can see that the proportion of the variance explained by day is unlikely to be less than a half and could be as high as three quarters.

Linear Mixed Model: Random effect hypothesis testing

Likelihood Ratio Test

Linear Mixed Model: Random effect hypothesis testing

Likelihood Ratio Test

We might also like to assess the significance of a variance component; how likely would we have been to get a estimate that large had the null-hypothesis (the variance is zero) been true. The most commonly used technique is to perform a likelihood-ratio test, and we used the logic of this test to understand the profile likelihood.

Linear Mixed Model: Random effect hypothesis testing

Likelihood Ratio Test

```
> anova(traffic_m6, traffic_m1)

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

           npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
traffic_m1    5 1314.5 1330.6 -652.27   1304.5
traffic_m6    6 1268.9 1288.2 -628.44   1256.9 47.656  1  5.081e-12
```


Linear Mixed Model: Random effect hypothesis testing

Likelihood Ratio Test

> anova(traffic_m6, traffic_m1)

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

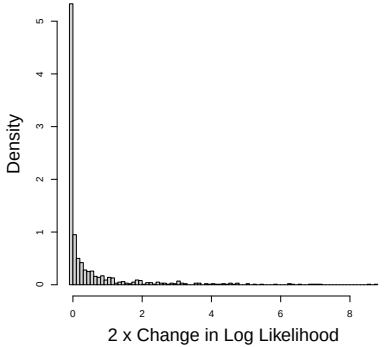
           npar AIC BIC logLik deviance Chisq Df Pr(>Chisq)
traffic_m1 5 1314.5 1330.6 -652.27 1304.5
traffic_m6 6 1268.9 1288.2 -628.44 1256.9 47.656 1 5.081e-12

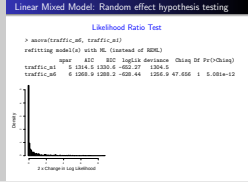
If we pass the function `anova` the model with day effects fitted `traffic_m6` and the simpler model with only fixed effects `traffic_m1` a likelihood-ratio test is performed by default. You can see that the day effects are highly significant. However, in this instance the approximation that underlies the likelihood-ratio test breaks down and is actually conservative.

Linear Mixed Model: Random effect hypothesis testing

Likelihood Ratio Test

```
> anova(traffic_m6, traffic_m1)
refitting model(s) with ML (instead of REML)
           npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
traffic_m1    5 1314.5 1330.6 -652.27   1304.5
traffic_m6    6 1268.9 1288.2 -628.44   1256.9 47.656  1  5.081e-12
```



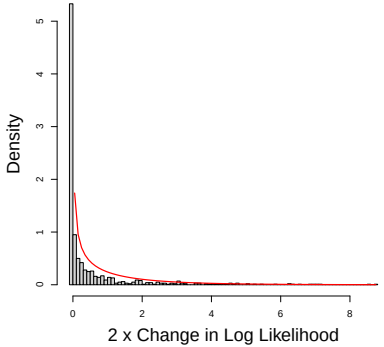


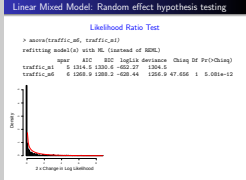
Here, I have simulated 1000 data sets under the null model (the variance of the day effects is zero), refitted both `traffic_m1` and `traffic_m6` to the data, and calculated twice the difference in their log-likelihoods. The likelihood ratio test, approximates this distribution as a chi-squared distribution with 1 degree of freedom (because the models differ by 1 parameter)

Linear Mixed Model: Random effect hypothesis testing

Likelihood Ratio Test

```
> anova(traffic_m6, traffic_m1)
refitting model(s) with ML (instead of REML)
          npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
traffic_m1    5 1314.5 1330.6 -652.27   1304.5
traffic_m6    6 1268.9 1288.2 -628.44   1256.9 47.656  1  5.081e-12
```





This solid red line is the probability density function for this chi-sqaured distribution, and you can see it's not a good approximation to the null distribution. The reason for this is that the null model is on the edge of the parameter space defined by the model to be tested; the day variance has to be positive - you can't have negative variances - and so a value of zero is on the edge of the allowable space. It is quite easy to understand why this issue upsets things in this instance. When we simulate random observations from the null model, observations on the same day are as likely to be similar as they are dissimilar. If they are on average similar, the estimate of the variance will be positive. If they are on average disimilar the estimate of the variance would like to go negative[1]. However, the variance is constrained to be positive and so in this instance an estimate of zero is returned and the change in likelihood is exactly zero.

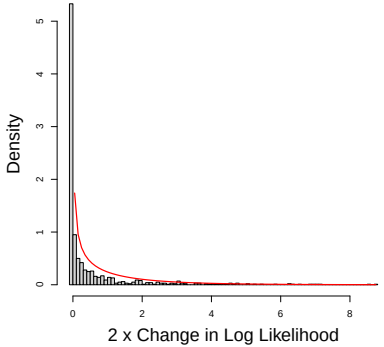
[1] Some programs (not `lmer`) will actually allow negative estimates of the variance parameter. Some people take issue with this because a variance has to be positive. However, earlier, one of the ways in which we tried to understand mixed models was thinking about the variance parameter as measuring the *covariance* between observations within a group. Covariances can be negative - perhaps observations within a group are more dissimilar than you expect by chance (imagine competing chicks within nests) - and so I don't take issue with negative estimates.

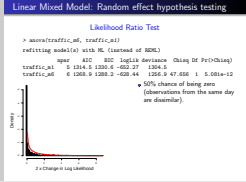
Linear Mixed Model: Random effect hypothesis testing

Likelihood Ratio Test

```
> anova(traffic_m6, traffic_m1)
refitting model(s) with ML (instead of REML)
           npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
traffic_m1    5 1314.5 1330.6 -652.27   1304.5
traffic_m6    6 1268.9 1288.2 -628.44   1256.9 47.656  1  5.081e-12
```

- 50% chance of being zero (observations from the same day are dissimilar).



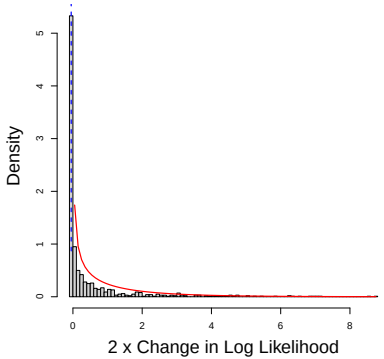


This should happen in 50% of cases and so you should get a spike at zero

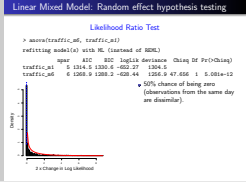
Linear Mixed Model: Random effect hypothesis testing

Likelihood Ratio Test

```
> anova(traffic_m6, traffic_m1)

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

           npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
traffic_m1    5 1314.5 1330.6 -652.27   1304.5
traffic_m6    6 1268.9 1288.2 -628.44   1256.9 47.656  1  5.081e-12
```



- 50% chance of being zero (observations from the same day are dissimilar).

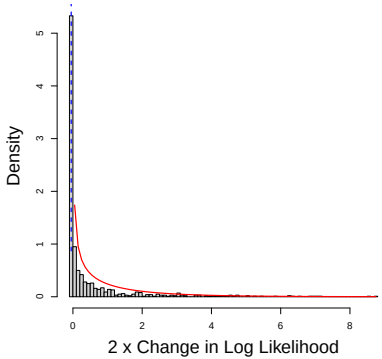


containing 50% of the probability, which I've represented with this blue line.

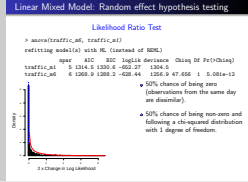
Linear Mixed Model: Random effect hypothesis testing

Likelihood Ratio Test

```
> anova(traffic_m6, traffic_m1)

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

           npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
traffic_m1    5 1314.5 1330.6 -652.27   1304.5
traffic_m6    6 1268.9 1288.2 -628.44   1256.9 47.656  1  5.081e-12
```



- 50% chance of being zero (observations from the same day are dissimilar).
- 50% chance of being non-zero and following a chi-squared distribution with 1 degree of freedom.

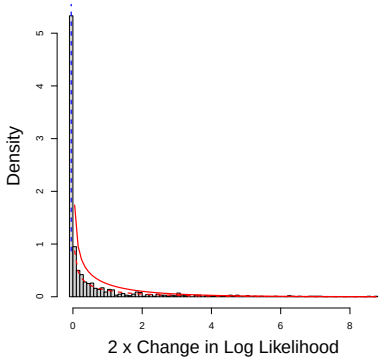


For the other 50%, their distribution will be chi-squared (with 1 degree of freedom), and so we can take our original chi-square probability density function (which had an area of 1)

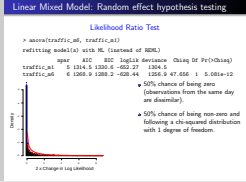
Linear Mixed Model: Random effect hypothesis testing

Likelihood Ratio Test

```
> anova(traffic_m6, traffic_m1)

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

           npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
traffic_m1    5 1314.5 1330.6 -652.27   1304.5
traffic_m6    6 1268.9 1288.2 -628.44   1256.9 47.656  1  5.081e-12
```



- 50% chance of being zero (observations from the same day are dissimilar).
- 50% chance of being non-zero and following a chi-squared distribution with 1 degree of freedom.

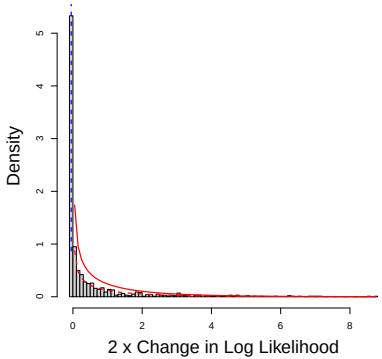


and halve its area, representing the fact that it should only describe 50% of the outcomes. I've plotted this as a dashed red line, and you can see that our new distribution which is a 50:50 mixture of a point mass at zero (blue-dotted) and a chi-square (red-dotted) fits the null distribution very well.

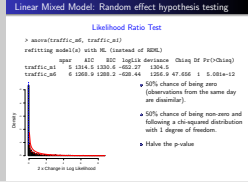
Linear Mixed Model: Random effect hypothesis testing

Likelihood Ratio Test

```
> anova(traffic_m6, traffic_m1)
refitting model(s) with ML (instead of REML)
           npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
traffic_m1    5 1314.5 1330.6 -652.27   1304.5
traffic_m6    6 1268.9 1288.2 -628.44   1256.9 47.656  1  5.081e-12
```



- 50% chance of being zero (observations from the same day are dissimilar).
- 50% chance of being non-zero and following a chi-squared distribution with 1 degree of freedom.
- Halve the p-value

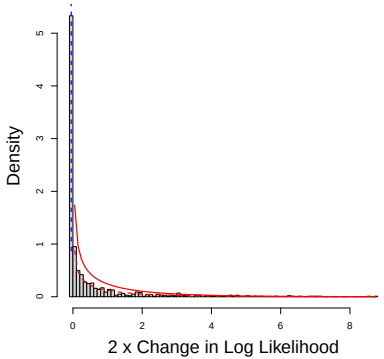


Practically, you can get the correct p-value by simply dividing the p-value returned by anova by half. In this instance it makes little substantive difference (5.080614e-12 and 2.540307e-12 are both very small).

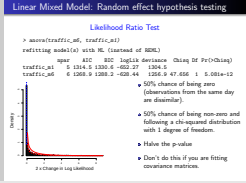
Linear Mixed Model: Random effect hypothesis testing

Likelihood Ratio Test

```
> anova(traffic_m6, traffic_m1)
refitting model(s) with ML (instead of REML)
           npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
traffic_m1    5 1314.5 1330.6 -652.27   1304.5
traffic_m6    6 1268.9 1288.2 -628.44   1256.9 47.656  1  5.081e-12
```



- 50% chance of being zero (observations from the same day are dissimilar).
- 50% chance of being non-zero and following a chi-squared distribution with 1 degree of freedom.
- Halve the p-value
- Don't do this if you are fitting covariance matrices.

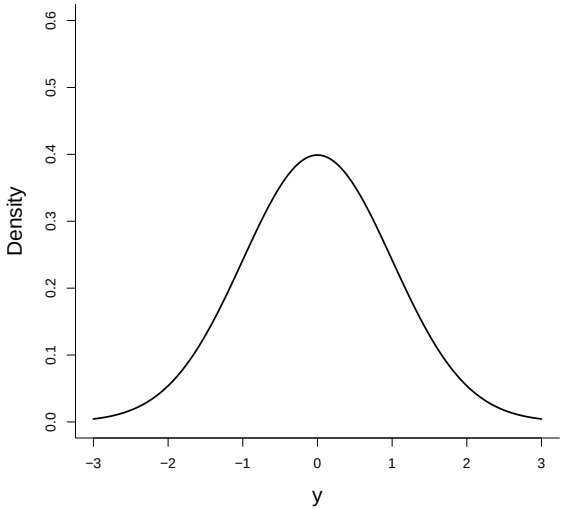


However, you have to remember you can only do this if you are testing a single variance - not something like a covariance matrix, which we'll come on to later. The other thing you might have noticed is that when performing a likelihood ratio test, `refitting model(s) with ML (instead of REML)` is printed to screen.

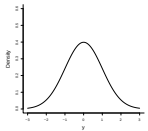
ML versus REML

I don't think it's massively important to understand the difference between a maximum likelihood estimate and a restricted maximum likelihood estimate, but I think I can give a quick sketch of what REML is trying to do and why, which might be useful.

ML versus REML

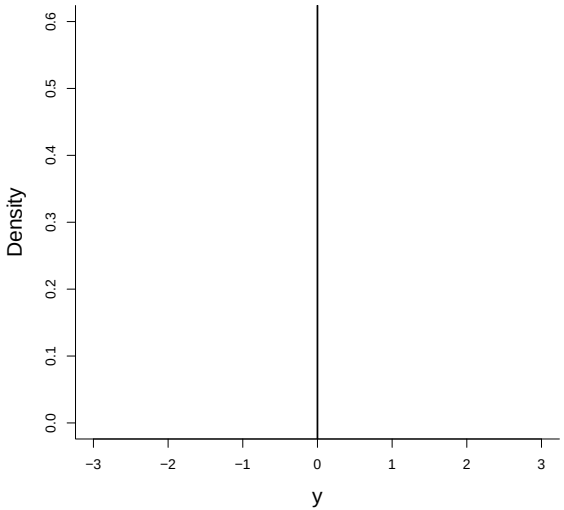


ML versus REML

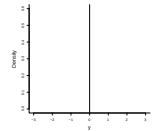


Imagine we have a normal distribution with a mean of zero and a variance of 1, and that twe would like to make inferences about the variance from some observations that were drawn from the distribution.

ML versus REML

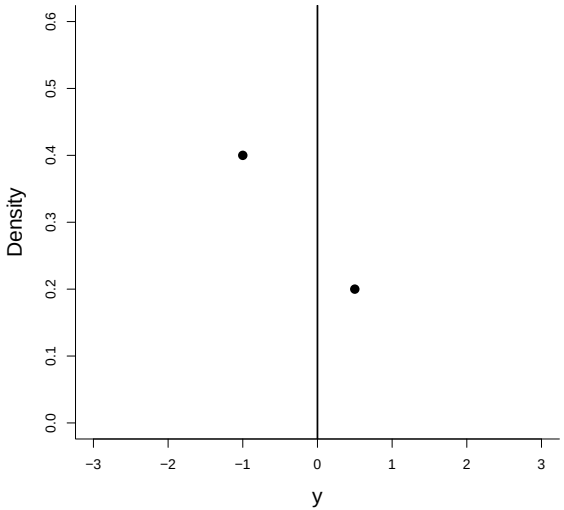


ML versus REML

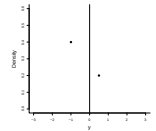


The distribution has a mean of zero

ML versus REML

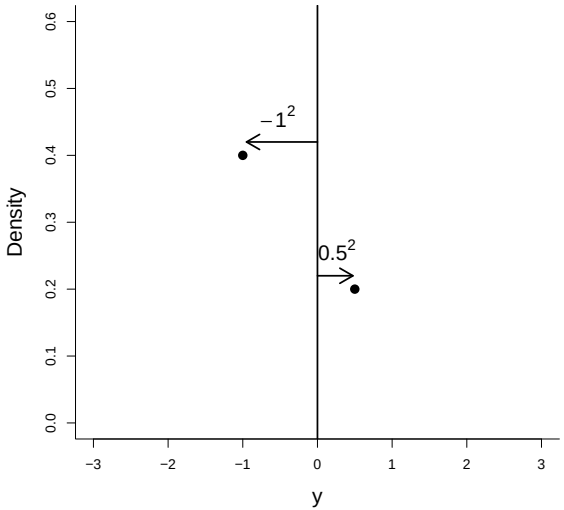


ML versus REML

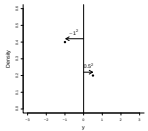


and let's say we only had two observations, one observation had a value of -1, and the second a value of 0.5.

ML versus REML

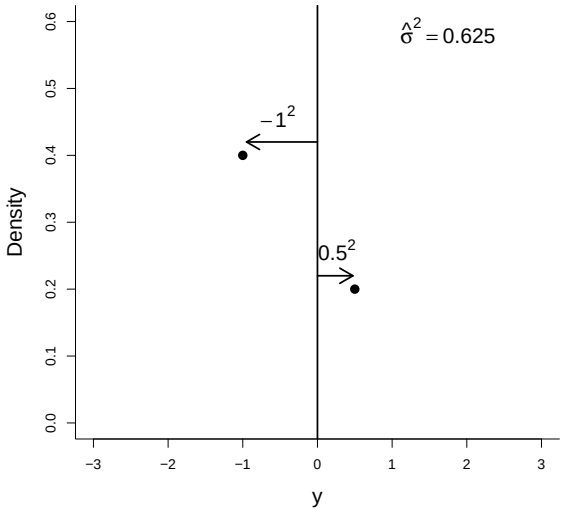


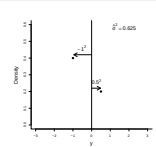
ML versus REML



Our best estimate of the variance, is to take the squared distance of each data point from the mean and average them.

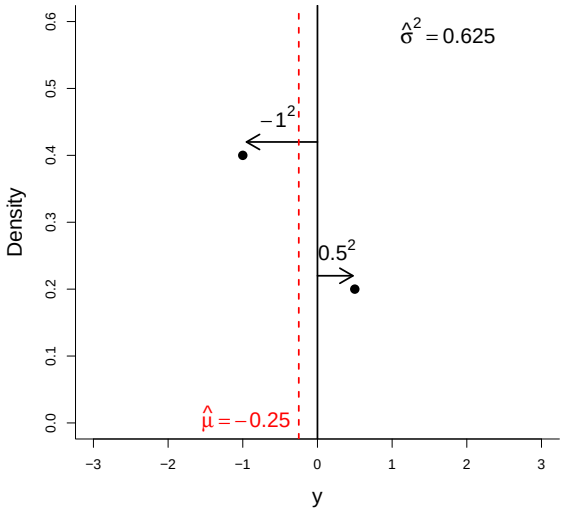
ML versus REML



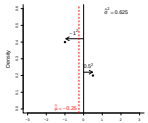


So $((-1)^2 + 0.5^2)/2 = 0.625$ is our best estimate of the variance. In fact this way of estimating the variance is unbiased; if we had sampled many pairs of observations, obtained an estimate of the variance in this way, and then looked at the average of our estimates it would coincide with the true value of 1. However, this is only true because we know the true mean (0) and so we can calculate the squared deviations from it. In practice, we usually have to estimate the mean and the variance from the same set of observations.

ML versus REML

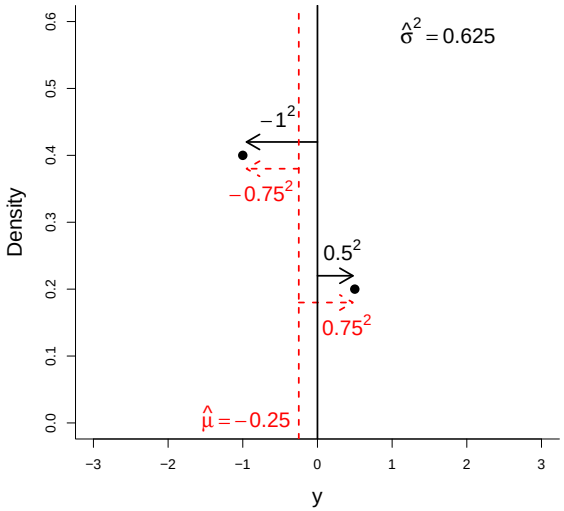


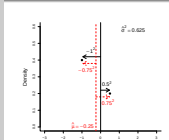
ML versus REML



Our maximum likelihood estimate of the mean is simply the mean of the observations, as we saw on the very first day ($(-1 + 0.5)/2 = -0.25$)

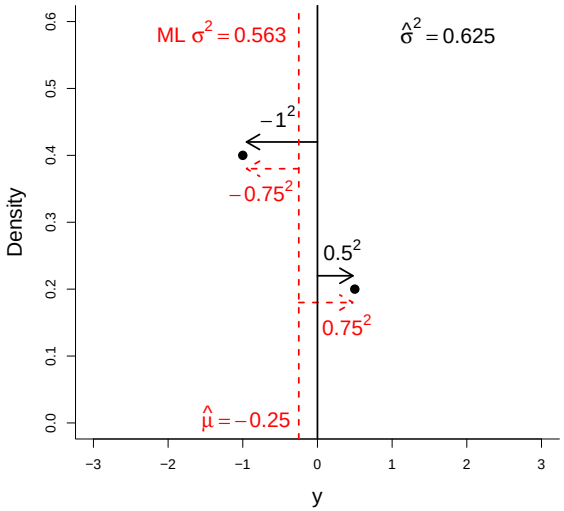
ML versus REML



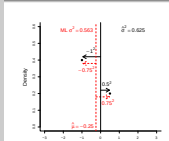


If we take the squared deviations of our data from our *estimate* of the mean, we can see that on average these squared distances are shorter than they are from the true mean. This has to be the case because the ML *estimate* of the mean is actually the value that minimises these squared distances, since it is the midpoint of all the data.

ML versus REML

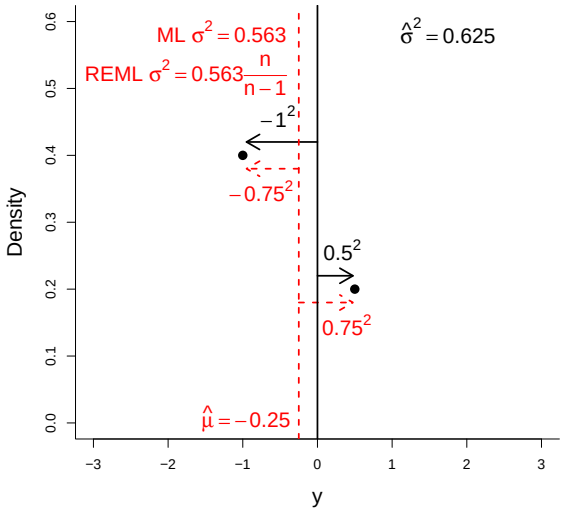


ML versus REML

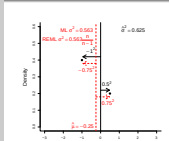


The ML estimate of the variance is the average of these squared distances, and because they are consistently shorter than they would be had we known the mean, they give a downwardly biased estimate of the true variance. Essentially, the ML estimator of the variance is downwardly biased because it fails to take into account the uncertainty we have about what the mean is. REML is a way of correcting the variance estimates for the estimation uncertainty in the fixed effect part of the model that determines the expected values. In this very simple scenario the REML estimator is well known.

ML versus REML



ML versus REML



We take our ML estimator of the variance and apply a finite-sample correction on $n/(n-1)$ which in this case (where $n = 2$) results in a doubling of the estimate. As sample sizes become large, the mean is very well estimated, and the impact of its estimation error on the estimate of the variance becomes trivial. In most cases therefore, the difference between the estimates obtained by REML versus ML are small.

On day 1 I claimed that I was getting ML estimates of the mean and variance for the grumpy/happy photos. In fact I was lying, the estimate of the residual variance was in fact a REML estimate, and `lm` is actually reporting the REML estimate of the residual variance. The functions `var` and `sd` are also returning REML estimates, although they are rarely referred to as such.

Linear Mixed Model: Likelihood-based tests

IMPORTANT

- A restricted likelihood (REML) cannot be compared with a standard likelihood (ML).

Linear Mixed Model: Likelihood-based tests

IMPORTANT

- A restricted likelihood (REML) cannot be compared with a standard likelihood (ML).

The REML algorithm deals with uncertainty in the fixed effects by essentially transforming the data prior to analysis. Because the data have changed, the likelihood reported using REML is not comparable to the likelihood reported using ML, because the likelihood is the probability of the (transformed) data given the model.

Linear Mixed Model: Likelihood-based tests

IMPORTANT

- A restricted likelihood (REML) cannot be compared with a standard likelihood (ML).
- Restricted likelihoods are not comparable if the fixed effects have changed.

Linear Mixed Model: Likelihood-based tests

IMPORTANT

- A restricted likelihood (REML) cannot be compared with a standard likelihood (ML).
- Restricted likelihoods are not comparable if the fixed effects have changed.

The transformation REML uses depends on the fixed effect structure of the model, and so if two models have been fitted to the same data using REML their likelihoods are not comparable if those models have different fixed effects (because the transformed data differ). If the models differ in their random effects only, then it is fine to compare the likelihoods obtaining using REML (or ML).

Linear Mixed Model: Likelihood-based tests

IMPORTANT

- A restricted likelihood (REML) cannot be compared with a standard likelihood (ML).
- Restricted likelihoods are not comparable if the fixed effects have changed.
- Likelihoods for the same models but from different programs may not be comparable.

Linear Mixed Model: Likelihood-based tests

IMPORTANT

- A restricted likelihood (REML) cannot be compared with a standard likelihood (ML).
- Restricted likelihoods are not comparable if the fixed effects have changed.
- Likelihoods for the same models but from different programs may not be comparable.

Also, because likelihoods only need to be known up to proportionality when model fitting (we just care about the relative differences in likelihood between models, or different parameter values within models) different programs fitting the same model using the same method may differ simply because they differ by some constant. Because of this you have to be quite careful comparing the likelihood of two models fitted by different functions.

Linear Mixed Model: Likelihood-based tests

IMPORTANT

- A restricted likelihood (REML) cannot be compared with a standard likelihood (ML).
- Restricted likelihoods are not comparable if the fixed effects have changed.
- Likelihoods for the same models but from different programs may not be comparable.
- Models compared via likelihood ratio tests must be *nested* (i.e the simpler model must be a special case of the more complex model)

Linear Mixed Model: Likelihood-based tests

IMPORTANT

- A restricted likelihood (REML) cannot be compared with a standard likelihood (ML).
- Restricted likelihoods are not comparable if the fixed effects have changed.
- Likelihoods for the same models but from different programs may not be comparable.
- Models compared via likelihood ratio tests must be *nested* (i.e the simpler model must be a special case of the more complex model)

Finally, when testing whether a parameter is significant or not using likelihood ratio tests, you have to make sure that the simpler model is nested (i.e. is a special case) of the full model. In our examples this has been true, the full model has a variance to be estimated, and the full model would coincide with the null model if the variance had been estimated to be exactly zero. However, if we had fitted two models, let's say one with day effects fitted as random but no year fixed year effect, and another with year fitted but not day, then we could not use the likelihood ratio test. This is because they are not nested; if we thought of our first model as the null model and it happened to return an estimate of the day variance as 1, the second model could never coincide with this 'null' because the variance of the day effects is implicitly zero.

Linear Mixed Model: Likelihood-based tests

IMPORTANT

- A restricted likelihood (REML) cannot be compared with a standard likelihood (ML).
- Restricted likelihoods are not comparable if the fixed effects have changed.
- Likelihoods for the same models but from different programs may not be comparable.
- Models compared via likelihood ratio tests must be *nested* (i.e the simpler model must be a special case of the more complex model)

Many functions, such as `anova` now check but if you are doing things 'by hand' you need to be careful.

Linear Mixed Model: Likelihood-based tests

IMPORTANT

- A restricted likelihood (REML) cannot be compared with a standard likelihood (ML).
- Restricted likelihoods are not comparable if the fixed effects have changed.
- Likelihoods for the same models but from different programs may not be comparable.
- Models compared via likelihood ratio tests must be nested (i.e the simpler model must be a special case of the more complex model)

Many functions, such as anova now check but if you are doing things 'by hand' you need to be careful.

In many cases, R now checks for these issues and either issues an error or refits the models so the likelihoods are comparable. In our example, we tried to compare a REML model estimated in `lmer` (`traffic_m6`) with a fixed effects model estimated in `lm` (`traffic_m1`). The function `anova` has refitted the REML model as ML, and has also checked to make sure the two functions, `lmer` and `lm` do not differ by some constant. However, in some cases you might need to perform these sorts of test 'by hand', and so it is good to be aware of them.

Are they fixed or random?

When you treat an effect as random you are saying that the *magnitude* of the coefficient associated with a level of a factor is informative about the likely *magnitude* of the coefficients at other levels. In most cases it is hard to see why this wouldn't be true whenever you have a factor with multiple levels.

Are they fixed or random?

```
> head(BTtarsus, 2)
```

	tarsus_mm	bird_id	sex	year	nest_orig	nest_rear	day_hatch
1	17.2	L298904	F	2011	11_A9	11_A9	0
2	17.6	L298903	M	2011	11_A9	11_A9	0

Are they fixed or random?

> head(BTtarsus, 2)

tarsus_mm bird_id sex year nest_orig nest_rear day_hatch
1 17.2 L298904 F 2011 11_A9 11_A9 0
2 17.6 L298903 M 2011 11_A9 11_A9 0

For example, these are a subset of the data we have been collecting on blue tits on the outskirts of Edinburgh. We have measured the length of the tarsus bone on many birds, we've genotyped them so we know whether they're male or female, and we've been doing this for several years now (The data we'll analyse are from 4 years). The final column is how long did a chick hatch after the first chick in its nest. If it's 0 it hatched on the same day, if it's a one it hatched the day after. We've also recorded the nest in which they were laid in as an egg (`nest_orig`), and the nest they were reared in after being laid (`nest_rear`) - in many cases these are different because we move eggs between nests, but we won't worry about nest effects today.

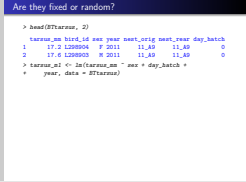
Are they fixed or random?

```
> head(BTtarsus, 2)
```

	tarsus_mm	bird_id	sex	year	nest_orig	nest_rear	day_hatch
1	17.2	L298904	F	2011	11_A9	11_A9	0
2	17.6	L298903	M	2011	11_A9	11_A9	0

```
> tarsus_m1 <- lm(tarsus_mm ~ sex + day_hatch +
+     year, data = BTtarsus)
```


Are they fixed or random?



We could imagine fitting this linear model where we have 1 intercept, 1 sex effect (the difference between males and females) and 1 day_hatch effect since we have treated it as continuous. However, we have 4 years of data and so 3 estimated year effects (the 3 differences from the base-line category, 2011) so we could have treated them as random, rather than fixed as here. People often argue over whether year effects should be treated as fixed or random. People often say that years haven't been sampled at random and so they cant be random effects, but as we've seen, this argument shows a deep misunderstanding of what a random effect is (random isn't referring to *years* being *sampled at random*, but referring to the fact that we would like to treat year *effects* as *random variables* coming from a distribution.) Conceptually they're random effects; if someone had told me that the year effects in the 20th century had ranged from -0.1mm to 0.1mm, I would be inclined to shrink my estimate for one of my years had it been 0.5mm and it was based on a small sample size. However, this seems to conflict with a rule of thumb people use to choose whether something is fixed or random; if the factor has few levels treat the effects as fixed and if it has many treat the effects as random. We only have four years, and so in this example I would treat the year effects as fixed although conceptually I think they are random. The reason for this is that when a factor has few levels it usually means each level is associated with a lot of data and so shrinkage is minimal and the estimates from the two approaches are nearly identical. However, hypothesis testing in the fixed effect model is more robust.

Are they fixed or random?

```
> head(BTtarsus, 2)
```

	tarsus_mm	bird_id	sex	year	nest_orig	nest_rear	day_hatch
1	17.2	L298904	F	2011	11_A9	11_A9	0
2	17.6	L298903	M	2011	11_A9	11_A9	0

```
> tarsus_m1 <- lm(tarsus_mm ~ sex + day_hatch +
+     year, data = BTtarsus)
> tarsus_null <- update(tarsus_m1, . ~ . - year)
```


Are they fixed or random?

> head(BTtarsus, 2)

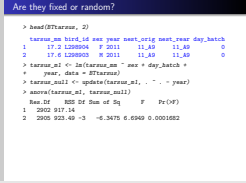
	tarsus_mm	bird_id	sex	year	nest_orig	nest_rear	day_hatch
1	17.2	L298904	F	2011	11_A9	11_A9	0
2	17.6	L298903	M	2011	11_A9	11_A9	0

> tarsus_m1 <- lm(tarsus_mm ~ sex + day_hatch +
+ year, data = BTtarsus)
> tarsus_null <- update(tarsus_m1, . ~ . - year)

For example, lets fit a null model which is identical to our model but without the year effects.

Are they fixed or random?

```
> head(BTtarsus, 2)

  tarsus_mm bird_id sex year nest_orig nest_rear day_hatch
1      17.2 L298904   F 2011     11_A9     11_A9         0
2      17.6 L298903   M 2011     11_A9     11_A9         0
> tarsus_m1 <- lm(tarsus_mm ~ sex + day_hatch +
+     year, data = BTtarsus)
> tarsus_null <- update(tarsus_m1, . ~ . - year)
> anova(tarsus_m1, tarsus_null)
  Res.Df    RSS Df Sum of Sq      F    Pr(>F)
1   2902 917.14
2   2905 923.49 -3   -6.3475 6.6949 0.0001682
```

anova performs an F-test, and tells us that the chance that all years have the same expected tarsus length (the 3 differences between 2011 and the remaining three years are all zero) is pretty low - about 1 in 5 or 6 thousand.

Are they fixed or random?

```
> head(BTtarsus, 2)

  tarsus_mm bird_id sex year nest_orig nest_rear day_hatch
1      17.2 L298904   F 2011     11_A9     11_A9         0
2      17.6 L298903   M 2011     11_A9     11_A9         0
> tarsus_m2 <- lmer(tarsus_mm ~ sex + day_hatch +
+     (1 | year), data = BTtarsus, REML = FALSE)
> tarsus_null <- update(tarsus_m1, . ~ . - year)
```


Are they fixed or random?

> head(BTtarsus, 2)

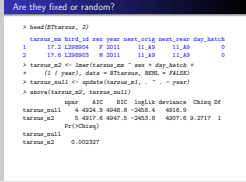
tarsus_mm bird_id sex year nest_orig nest_rear day_hatch
1 17.2 L298904 F 2011 11_A9 11_A9 0
2 17.6 L298903 M 2011 11_A9 11_A9 0

> tarsus_m2 <- lmer(tarsus_mm ~ sex + day_hatch +
+ (1 | year), data = BTtarsus, REML = FALSE)
> tarsus_null <- update(tarsus_m1, . ~ . - year)

We could also fit a similar model, but treat year as random effect rather than a fixed effect, and then compare this model to the same null model without any year effects in it.

Are they fixed or random?

```
> head(BTtarsus, 2)

  tarsus_mm bird_id sex year nest_orig nest_rear day_hatch
1      17.2 L298904   F 2011     11_A9     11_A9         0
2      17.6 L298903   M 2011     11_A9     11_A9         0
> tarsus_m2 <- lmer(tarsus_mm ~ sex + day_hatch +
+     (1 | year), data = BTtarsus, REML = FALSE)
> tarsus_null <- update(tarsus_m1, . ~ . - year)
> anova(tarsus_m2, tarsus_null)
            npar    AIC    BIC  logLik deviance  Chisq Df
tarsus_null    4 4924.9 4948.8 -2458.4   4916.9
tarsus_m2      5 4917.6 4947.5 -2453.8   4907.6 9.2717  1
            Pr(>Chisq)
tarsus_null
tarsus_m2     0.002327
```

It's significant, as before, but the p-value is much less extreme; only about 1 in 400 compared to 1 in 5 or 6 thousand. Even halving the p-value doesn't bring it close to the p-value from the fixed effect model. This seems surprising. The hypotheses been tested in the two models seem to be the same since all years being identical implies that the between-year variance is zero. It is tempting to believe that the mixed model is more robust because it shrinks the year effects back to a common mean and so their smaller magnitude results in a less extreme p-value. In fact, this isn't the case. The difference arises because there are few years and so the between-year variance is poorly estimated, and the approximation underlying a likelihood-ratio test for the variance is poor. In contrast, each year effect is well estimated and so the approximations underlying the fixed-effect tests should be good (in fact without other random effects in the model, the F-test as used is exact.)

Are they fixed or random?

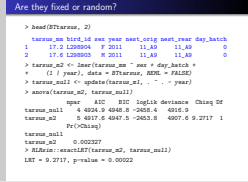
```
> head(BTtarsus, 2)
```

	tarsus_mm	bird_id	sex	year	nest_orig	nest_rear	day_hatch
1	17.2	L298904	F	2011	11_A9	11_A9	0
2	17.6	L298903	M	2011	11_A9	11_A9	0

```
> tarsus_m2 <- lmer(tarsus_mm ~ sex + day_hatch +
+     (1 | year), data = BTtarsus, REML = FALSE)
> tarsus_null <- update(tarsus_m1, . ~ . - year)
> anova(tarsus_m2, tarsus_null)
```

	npar	AIC	BIC	logLik	deviance	Chisq	Df	Pr(>Chisq)
tarsus_null	4	4924.9	4948.8	-2458.4	4916.9			
tarsus_m2	5	4917.6	4947.5	-2453.8	4907.6	9.2717	1	0.002327

```
> RLRsim::exactLRT(tarsus_m2, tarsus_null)
LRT = 9.2717, p-value = 0.00022
```

Rather than relying on the approximated distribution of the likelihood-ratio, we can generate the distribution of likelihoods under the two models using simulation, and assess how likely our actual likelihood-ratio really is. We can do this using the function `exactLRT` and we can see that the p-value for the random effect model and the fixed effect model are almost identical.

Are they fixed or random?

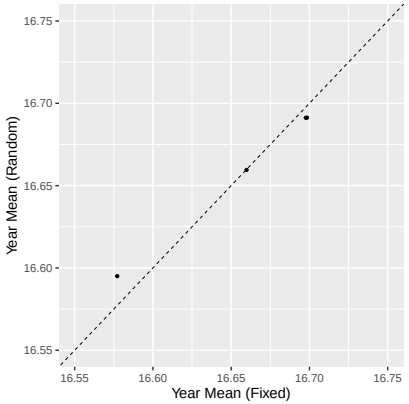
```
> fixef(tarsus_m2)[1]
(Intercept)
   16.65926
> ranef(tarsus_m2)[1]
$year
      (Intercept)
2011  0.0319286735
2012  0.0002950763
2013 -0.0642531743
2014  0.0320294245
```


Are they fixed or random?

> fixef(tarsus_m2)[1]
(Intercept)
16.65926
> ranef(tarsus_m2)[1]
$year
(Intercept)
2011 0.0319286735
2012 0.0002950763
2013 -0.0642531743
2014 0.0320294245

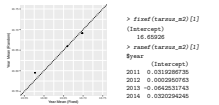
Usually we don't look at the random effects in a fixed effect model. If I'd fitted nest effects as random, I certainly wouldn't be interested in knowing what the 400 nest effects were I would simply be interested in who variable they were. However, we can get the random effects if we like using the function `ranef`. If we take the intercept and add it to each year effect we get the expected tarsus lengths in each year (for females that hatched on day 0).

Are they fixed or random?



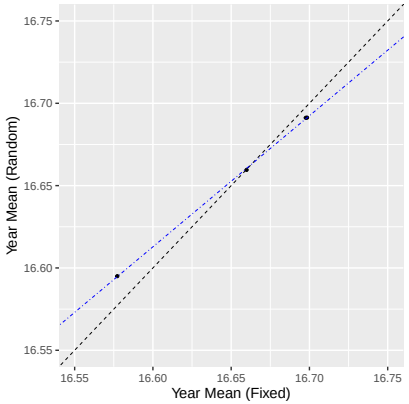

```
> fixef(tarsus_m2)[1]
(Intercept)
   16.65926
> ranef(tarsus_m2)[1]
$year
      (Intercept)
2011  0.0319286735
2012  0.0002950763
2013 -0.0642531743
2014  0.0320294245
```


Are they fixed or random?



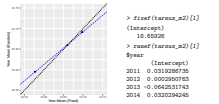
Here, I've plotted the random effect estimates (y-axis) against the fixed effect estimates (x-axis; in the fixed effect model 2011 is the intercept, and the other year effects are deviations from this) and you can see that they lie pretty close to the 1:1 line. 2011 and 2014 are obscured because their effects are very similar.

Are they fixed or random?



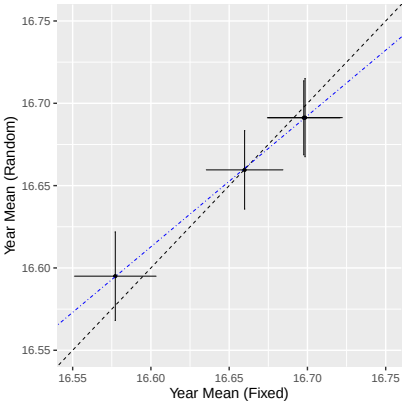

```
> fixef(tarsus_m2)[1]
(Intercept)
   16.65926
> ranef(tarsus_m2)[1]
$year
       (Intercept)
2011  0.0319286735
2012  0.0002950763
2013 -0.0642531743
2014  0.0320294245
```


Are they fixed or random?



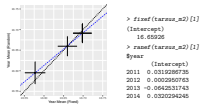
The slope is a little shallower because there is some shrinkage (the random effect estimates are closer together) but it's fairly minimal because there's a lot of information within a year.

Are they fixed or random?




```
> fixef(tarsus_m2)[1]
(Intercept)
   16.65926
> ranef(tarsus_m2)[1]
$year
      (Intercept)
2011  0.0319286735
2012  0.0002950763
2013 -0.0642531743
2014  0.0320294245
```


Are they fixed or random?



We can also plot the standard errors on these estimates, and we can see that again the standard errors associated with the different approaches are very similar. It really makes little difference if we treat year as fixed or random and there is little point agonising or arguing over it (as long as the issues associated with hypothesis testing in the random effect model are dealt with). I don't mean to imply that this will always be the case; if you have many levels, with few observations associated with each, it certainly won't be the case. However, often people worry about the distinction when they have few levels, and in these cases it often makes little difference unless sample sizes are tiny.