Douglas M. Bates

lme4: Mixed-effects modeling with R

January 18, 2010

Springer

Contents

List of Figures

Chapter 1
A Simple, Linear, Mixed-effects Model

In this book we describe the theory behind a type of statistical model called *mixed-effects* models and the practice of fitting and analyzing such models using the `lme4` package for R. These models are used in many different disciplines. Because the descriptions of the models can vary markedly between disciplines, we begin by describing what mixed-effects models are and by exploring a very simple example of one type of mixed model, the *linear mixed model*.

This simple example allows us to illustrate the use of the `lmer` function in the `lme4` package for fitting such models and analyzing the fitted model. Building from the example we describe the general form of linear mixed models that can be fit using `lmer`.

1.1 Mixed-effects Models

Mixed-effects models, like many other types of statistical models, describe a relationship between a *response* variable and some of the *covariates* that have been measured or observed along with the response. In mixed-effects models at least one of the covariates is a *categorical* covariate representing experimental or observational "units" in the data set. In the example from the chemical industry that is given in this chapter, the observational unit is the batch of an intermediate product used in production of a dye. In medical and social sciences the observational units are often the human or animal subjects in the study. In agriculture the experimental units may be the plots of land or the specific plants being studied.

In all of these cases the categorical covariate or covariates are observed at a set of discrete *levels*. We may use numbers, such as subject identifiers, to designate the particular levels that we observed but these numbers are simply labels. The important characteristic of a categorical covariate is that, at each

observed value of the response, the covariate takes on the value of one of a set of distinct levels.

Parameters associated with the particular levels of a covariate are sometimes called the "effects" of the levels. If the set of possible levels of the covariate is fixed and reproducible we model the covariate using *fixed-effects* parameters. If the levels that we observed represent a random sample from the set of all possible levels we incorporate *random effects* in the model.

There are two things to notice about this distinction between fixed-effects parameters and random effects. First, the names are misleading because the distinction between fixed and random is more a property of the levels of the categorical covariate than a property of the effects associated with them. Secondly, we distinguish between "fixed-effects parameters", which are indeed parameters in the statistical model, and "random effects", which, strictly speaking, are not parameters. As we will see shortly, random effects are unobserved random variables.

To make the distinction more concrete, suppose that we wish to model the annual reading test scores for students in a school district and that the covariates recorded with the score include a student identifier and the student's gender. Both of these are categorical covariates. The levels of the gender covariate, male and female, are fixed. If we consider data from another school district or we incorporate scores from earlier tests, we will not change those levels. On the other hand, the students whose scores we observed would generally be regarded as a sample from the set of all possible students whom we could have observed. Adding more data, either from more school districts or from results on previous or subsequent tests, will increase the number of distinct levels of the student identifier.

Mixed-effects models or, more simply, *mixed models* are statistical models that incorporate both fixed-effects parameters and random effects. Because of the way that we will define random effects, a model with random effects always includes at least one fixed-effects parameter. Thus, any model with random effects is a mixed model.

We characterize the statistical model in terms of two random variables: a q-dimensional vector of random effects represented by the random variable $\mathscr{B}$ and an n-dimensional response vector represented by the random variable $\mathscr{Y}$. We observe the value, $\mathbf{y}$, of $\mathscr{Y}$. We do not observe the value of $\mathscr{B}$.

When formulating the model we describe the unconditional distribution of $\mathscr{B}$ and the conditional distribution, $(\mathscr{Y}|\mathscr{B} = \mathbf{b})$. The descriptions of the distributions involve the form of the distribution and the values of certain parameters. We use the observed values of the response and the covariates to estimate these parameters and to make inferences about them.

That the big picture. Now let's make this more concrete by describing a particular, versatile class of mixed models called linear mixed models and by studying a simple example of such a model. First we will describe the data in the example.

1.2 The Dyestuff and Dyestuff2 Data

Models with random effects have been in use for a long time. The first edition of the classic book, *Statistical Methods in Research and Production*, edited by O.L. Davies, was published in 1947 and contained examples of the use of random effects to characterize batch-to-batch variability in chemical processes. The data from one of these examples are available as the `Dyestuff` data in the `lme4` package. In this section we describe and plot these data and introduce a second example, the `Dyestuff2` data, described in Box and Tiao [1973].

1.2.1 The Dyestuff *Data*

The `Dyestuff` data are described in Davies and Goldsmith [1972, Table 6.3, p. 131], the fourth edition of the book mentioned above, as coming from

> an investigation to find out how much the variation from batch to batch in the quality of an intermediate product (H-acid) contributes to the variation in the yield of the dyestuff (Naphthalene Black 12B) made from it. In the experiment six samples of the intermediate, representing different batches of works manufacture, were obtained, and five preparations of the dyestuff were made in the laboratory from each sample. The equivalent yield of each preparation as grams of standard colour was determined by dye-trial.

To access these data within R we must first attach the `lme4` package to our session using

```
> library(lme4)
```

Note that the `">"` symbol in the line shown is the prompt in R and not part of what the user types. The `lme4` package must be attached before any of the data sets or functions in the package can be used. If typing this line results in an error report stating that there is no package by this name then you must first install the package.

In what follows, we will assume that the `lme4` package has been installed and that it has been attached to the R session before any of the code shown has been run.

The `str` function in R provides a concise description of the structure of the data

```
> str(Dyestuff)
```

```
'data.frame':        30 obs. of  2 variables:
 $ Batch: Factor w/ 6 levels "A","B","C","D",..: 1 1 1 1 1 2 2 2 2 2 ...
 $ Yield: num  1545 1440 1440 1520 1580 ...
```

from which we see that it consists of 30 observations of the `Yield`, the response variable, and of the covariate, `Batch`, which is a categorical variable stored as a `factor` object. If the labels for the factor levels are arbitrary, as they are

here, we will use letters instead of numbers for the labels. That is, we label the batches as `"A"` through `"F"` rather than `"1"` through `"6"`. When the labels are letters it is clear that the variable is categorical. When the labels are numbers a categorical covariate can be mistaken for a numeric covariate, with unintended consequences.

It is a good practice to apply `str` to any data frame the first time you work with it and to check carefully that any categorical variables are indeed represented as factors.

The data in a data frame are viewed as a table with columns corresponding to variables and rows to observations. The functions `head` and `tail` print the first or last few rows (the default value of "few" happens to be 6 but we can specify another value if we so choose)

```
> head(Dyestuff)
```

```
  Batch Yield
1     A  1545
2     A  1440
3     A  1440
4     A  1520
5     A  1580
6     B  1540
```

or we could ask for a `summary` of the data

```
> summary(Dyestuff)
```

```
 Batch     Yield
 A:5   Min.   :1440
 B:5   1st Qu.:1469
 C:5   Median :1530
 D:5   Mean   :1528
 E:5   3rd Qu.:1575
 F:5   Max.   :1635
```

Although the `summary` does show us an important property of the data, namely that there are exactly 5 observations on each batch — a property that we will describe by saying that the data are *balanced* with respect to `Batch` — we usually learn much more about the structure of such data from plots like Fig. 1.1 than we can from numerical summaries.

In Fig. 1.1 we can see that there is considerable variability in yield, even for preparations from the same batch, but there is also noticeable batch-to-batch variability. For example, four of the five preparations from batch F provided lower yields than did any of the preparations from batches C and E.

This plot, and essentially all the other plots in this book, were created using Deepayan Sarkar's `lattice` package for R. In Sarkar [2008] he describes how one would create such a plot. Because this book was created using Sweave [Leisch, 2002], the exact code used to create the plot, as well as the code for all the other figures and calculations in the book, is available on the web site for the book. In section **??** we review some of the principles of lattice

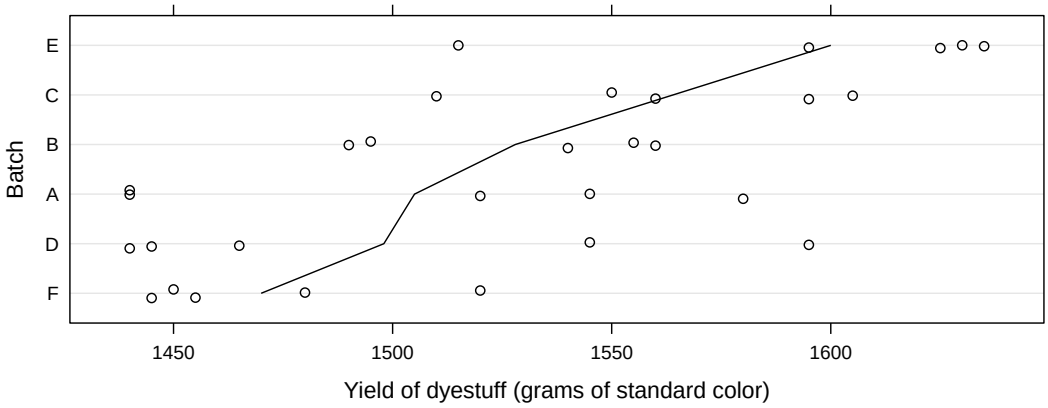


Fig. 1.1 Yield of dyestuff (Napthalene Black 12B) for 5 preparations from each of 6 batches of an intermediate product (H-acid). The line joins the mean yields from the batches, which have been ordered by increasing mean yield. The vertical positions are "jittered" slightly to avoid over-plotting. Notice that the lowest yield for batch A was observed for two distinct preparations from that batch.

graphics, such as reordering the levels of the `Batch` factor by increasing mean response, that enhance the informativeness of the plot. At this point we will concentrate on the information conveyed by the plot and not on how the plot is created.

In section 1.3.1 we will use mixed models to quantify the variability in yield between batches. For the time being let us just note that the particular batches used in this experiment are a selection or sample from the set of all batches that we wish to consider. Furthermore, the extent to which one particular batch tends to increase or decrease the mean yield of the process — in other words, the "effect" of that particular batch on the yield — is not as interesting to us as is the extent of the variability between batches. For the purposes of designing, monitoring and controlling a process we want to predict the yield from future batches, taking into account the batch-to-batch variability and the within-batch variability. Being able to estimate the extent to which a particular batch in the past increased or decreased the yield is not usually an important goal for us. We will model the effects of the batches as random effects rather than as fixed-effects parameters.

1.2.2 The `Dyestuff2` *Data*

The `Dyestuff2` data are simulated data presented in Box and Tiao [1973, Table 5.1.4, p. 247] where the authors state

> These data had to be constructed for although examples of this sort undoubtedly occur in practice they seem to be rarely published.

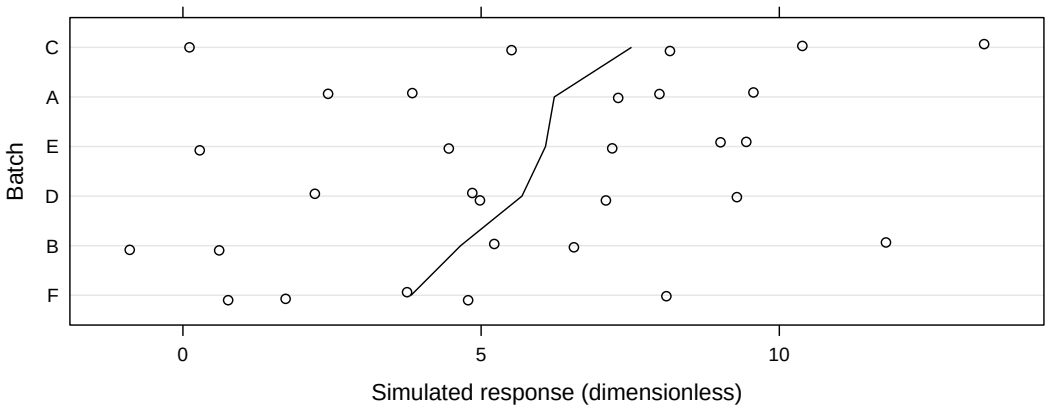


Fig. 1.2 Simulated data presented in Box and Tiao [1973] with a structure similar to that of the `Dyestuff` data. These data represent a case where the batch-to-batch variability is small relative to the within-batch variability.

The structure and summary

```
> str(Dyestuff2)

'data.frame':        30 obs. of  2 variables:
 $ Batch: Factor w/ 6 levels "A","B","C","D",..: 1 1 1 1 1 2 2 2 2 2 ...
 $ Yield: num  7.3 3.85 2.43 9.57 7.99 ...

> summary(Dyestuff2)

 Batch     Yield
 A:5   Min.   :-0.892
 B:5   1st Qu.: 2.765
 C:5   Median : 5.365
 D:5   Mean   : 5.666
 E:5   3rd Qu.: 8.151
 F:5   Max.   :13.434
```

are intentionally similar to those of the `Dyestuff` data. As can be seen in Fig. 1.2 the batch-to-batch variability in these data is small compared to the within-batch variability. In some approaches to mixed models it can be difficult to fit models to such data. Paradoxically, small "variance components" can be more difficult to estimate than large variance components.

The methods we will present are not compromised when estimating small variance components.

1.3 Fitting Linear Mixed Models

Before we formally define a linear mixed model, let's go ahead and fit models to these data sets using `lmer`. Like most model-fitting functions in R, `lmer`

takes, as its first two arguments, a *formula* specifying the model and the *data* with which to evaluate the formula. This second argument, `data`, is optional but recommended. It is usually the name of a data frame, such as those we examined in the last section. Throughout this book all model specifications will be given in this formula/data format.

We will explain the structure of the formula after we have considered an example.

1.3.1 A Model For the `Dyestuff` *Data*

We fit a model to the `Dyestuff` data allowing for an overall level of the `Yield` and for an additive random effect for each level of `Batch`

```
> fm1 <- lmer(Yield ~ 1 + (1|Batch), Dyestuff)
> print(fm1)

Linear mixed model fit by REML
Formula: Yield ~ 1 + (1 | Batch)
   Data: Dyestuff
 REML
319.7

Random effects:
 Groups   Name        Variance Std.Dev.
 Batch    (Intercept) 1764.0   42.001
 Residual             2451.3   49.510
Number of obs: 30, groups: Batch, 6

Fixed effects:
            Estimate Std. Error t value
(Intercept)  1527.50      19.38    78.8
```

In the first line we call the `lmer` function to fit a model with formula

```
Yield ~ 1 + (1 | Batch)
```

applied to the `Dyestuff` data and assign the result to the name `fm1`. (The name is arbitrary. I happen to use names that start with `fm`, indicating "fitted model".)

As is customary in R, there is no output shown after this assignment. We have simply saved the fitted model as an object named `fm1`. In the second line we display some information about the fitted model by applying `print` to `fm1`. In later examples we will condense these two steps into one but here it helps to emphasize that we save the result of fitting a model then apply various *extractor* functions to the fitted model to get a brief summary of the model fit or to obtain the values of some of the estimated quantities.

1.3.1.1 Details of the Printed Display

The printed display of a model fit with `lmer` has four major sections: a description of the model that was fit, some statistics characterizing the model fit, a summary of properties of the random effects and a summary of the fixed-effects parameter estimates. We consider each of these sections in turn.

The description section states that this is a linear mixed model in which the parameters have been estimated as those that minimize the REML criterion (explained in section **??**). The `formula` and `data` arguments are displayed for later reference. If other, optional arguments affecting the fit, such as a `subset` specification, were used, they too will be displayed here.

For models fit by the REML criterion the only statistic describing the model fit is the value of the REML criterion itself. An alternative set of parameter estimates, the maximum likelihood estimates, are obtained by specifying the optional argument `REML = FALSE`.

```
> (fm1ML <- lmer(Yield ~ 1 + (1|Batch), Dyestuff, REML = FALSE))

Linear mixed model fit by maximum likelihood
Formula: Yield ~ 1 + (1 | Batch)
   Data: Dyestuff
   AIC   BIC logLik deviance
 333.3 337.5 -163.7    327.3

Random effects:
 Groups   Name        Variance Std.Dev.
 Batch    (Intercept) 1388.3   37.26
 Residual             2451.3   49.51
Number of obs: 30, groups: Batch, 6

Fixed effects:
            Estimate Std. Error t value
(Intercept)  1527.50      17.69   86.33
```

(Notice that this code fragment also illustrates a way to condense the assignment and the display of the fitted model into a single step. The redundant set of parentheses surrounding the assignment causes the result of the assignment to be displayed. We will use this device often in what follows.)

The display of a model fit by maximum likelihood provides several other model-fit statistics such as Akaike's Information Criterion (`AIC`) [Sakamoto et al., 1986], Schwarz's Bayesian Information Criterion (`BIC`) [Schwarz, 1978], the log-likelihood (`logLik`) at the parameter estimates, and the deviance (negative twice the log-likelihood) at the parameter estimates. These are all statistics related to the model fit and are used to compare different models fit to the same data.

At this point the important thing to note is that the default estimation criterion is the REML criterion. Generally the REML estimates of variance components are preferred to the ML estimates. However, when comparing

models it is safest to refit all the models using the maximum likelihood criterion. We will discuss comparisons of model fits later in section **??**.

The third section is the table of estimates of parameters associated with the random effects. There are two sources of variability in the model we have fit, a batch-to-batch variability in the level of the response and the residual or per-observation variability — also called the within-batch variability. The name "residual" is used in statistical modeling to denote the part of the variability that cannot be explained or modeled with the other terms. It is the variation in the observed data that is "left over" after we have determined the estimates of the parameters in the other parts of the model.

Some of the variability in the response is associated with the fixed-effects terms. In this model there is only one such term, labeled as the `(Intercept)`. The name "intercept", which is better suited to models based on straight lines written in a slope/intercept form, should be understood to represent an overall "typical" or mean level of the response in this case. (In case you are wondering about the parentheses around the name, they are included so that you can't accidentally create a variable with a name that conflicts with this name.) The line labeled `Batch` in the random effects table shows that the random effects added to the `(Intercept)` term, one for each level of the `Batch` factor, are modeled as random variables whose unconditional variance is estimated as 1764.05 gm.^{2} in the REML fit and as 1388.33 gm.^{2} in the ML fit. The corresponding standard deviations are 42.00 gm. for the REML fit and 37.26 gm. for the ML fit.

Note that the last column in the random effects summary table is the estimate of the variability expressed as a standard deviation rather than as a variance. These are provided because it is usually easier to visualize standard deviations, which are on the scale of the response, than it is to visualize the magnitude of a variance. The values in this column are a simple re-expression (the square root) of the estimated variances. Do not confuse them with the standard errors of the variance estimators, which are not given here. In section 1.5 we explain why we do not provide standard errors of variance estimates.

The line labeled `Residual` in this table gives the estimate of the variance of the residuals (also in gm.^{2}) and its corresponding standard deviation. For the REML fit the estimated standard deviation of the residuals is 49.51 gm. and for the ML fit it is also 49.51 gm. (Generally these estimates do not need to be equal. They happen to be equal in this case because of the simple model form and the balanced data set.)

The last line in the random effects table states the number of observations to which the model was fit and the number of levels of any "grouping factors" for the random effects. In this case we have a single random effects term, `(1|Batch)`, in the model formula and the grouping factor for that term is `Batch`. There will be a total of six random effects, one for each level of `Batch`.

The final part of the printed display gives the estimates and standard errors of any fixed-effects parameters in the model. The only fixed-effects

term in the model formula is the `1`, denoting a constant which, as explained above, is labeled as `(Intercept)`. For both the REML and the ML estimation criterion the estimate of this parameter is 1527.5 gm. (equality is again a consequence of the simple model and balanced data set). The standard error of the intercept estimate is 19.38 gm. for the REML fit and 17.69 gm. for the ML fit.

1.3.2 A Model For the `Dyestuff2` *Data*

Fitting a similar model to the `Dyestuff2` data produces an estimate $\widehat{\sigma}_1 = 0$ in both the REML

```
> (fm2 <- lmer(Yield ~ 1 + (1|Batch), Dyestuff2))
```

```
Linear mixed model fit by REML
Formula: Yield ~ 1 + (1 | Batch)
   Data: Dyestuff2
 REML
161.8

Random effects:
 Groups   Name        Variance Std.Dev.
 Batch    (Intercept)  0.000   0.0000
 Residual             13.806   3.7157
Number of obs: 30, groups: Batch, 6

Fixed effects:
            Estimate Std. Error t value
(Intercept)   5.6656     0.6784   8.352
```

and the ML fits.

```
> (fm2ML <- update(fm2, REML = FALSE))
```

```
Linear mixed model fit by maximum likelihood
Formula: Yield ~ 1 + (1 | Batch)
   Data: Dyestuff2
   AIC   BIC logLik deviance
 168.9 173.1 -81.44    162.9

Random effects:
 Groups   Name        Variance Std.Dev.
 Batch    (Intercept)  0.000   0.0000
 Residual             13.346   3.6532
Number of obs: 30, groups: Batch, 6

Fixed effects:
            Estimate Std. Error t value
(Intercept)    5.666      0.667   8.494
```

(Note the use of the `update` function to re-fit a model changing some of the arguments. In a case like this, where the call to fit the original model is not

very complicated, the use of `update` is not that much simpler than repeating the original call to `lmer` with extra arguments. For complicated model fits it can be.)

An estimate of 0 for σ_1 does not mean that there is no variation between the groups. Indeed Fig. 1.2 shows that there is some small amount of variability between the groups. The estimate, $\widehat{\sigma}_1 = 0$, simply indicates that the level of "between-group" variability is not sufficient to warrant incorporating random effects in the model.

The important point to take away from this example is that we must allow for the estimates of variance components to be zero. We describe such a model as being degenerate, in the sense that it corresponds to a linear model in which we have removed the random effects associated with `Batch`. Degenerate models can and do occur in practice. Even when the final fitted model is not degenerate, we must allow for such models when determining the parameter estimates through numerical optimization.

To reiterate, the model `fm2` corresponds to the linear model

```
> summary(fm2a <- lm(Yield ~ 1, Dyestuff2))

Call:
lm(formula = Yield ~ 1, data = Dyestuff2)

Residuals:
    Min      1Q  Median      3Q     Max
-6.5576 -2.9006 -0.3006  2.4854  7.7684

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   5.6656     0.6784   8.352 3.32e-09

Residual standard error: 3.716 on 29 degrees of freedom
```

because the random effects are inert, in the sense that they have a variance of zero, and can be removed.

Notice that the estimate of σ from the linear model (called the `Residual standard error` in the output corresponds to the estimate in the REML fit (`fm2`) but not that from the ML fit (`fm2ML`). The fact that the REML estimates of variance components generalize the estimate of the variance used in linear models, in the sense that they correspond in the degenerate case, is part of the motivation for the use of the REML criterion for fitting mixed-effects models.

1.3.3 Further Assessment of the Fitted Models

The parameter estimates in a statistical model represent our "best guess" at the unknown values of the model parameters and, as such, are important

results in statistical modeling. However, they are not the whole story. Statistical models characterize the variability in the data and we must assess the effect of this variability on the parameter estimates and on the precision of predictions made from the model.

In section 1.5 we introduce a method of assessing variability in parameter estimates using the "profiled deviance" and in section 1.6 we show methods of characterizing the conditional distribution of the random effects given the data. Before we get to these sections, however, we should state in some detail the probability model for linear mixed-effects and establish some definitions and notation. In particular, before we can discuss profiling the deviance, we should define the deviance. We do that in the next section.

1.4 The Linear Mixed-effects Probability Model

In explaining some of parameter estimates related to the random effects we have used terms such as "unconditional distribution" from the theory of probability. Before proceeding further we should clarify the linear mixed-effects probability model and define several terms and concepts that will be used throughout the book.

1.4.1 Definitions and Results

In this section we provide some definitions and formulas without derivation and with minimal explanation, so that we can use these terms in what follows. In Chapter **??** we revisit these definitions providing derivations and more explanation.

As mentioned in section 1.1, a mixed model incorporates two random variables: $\mathscr{B}$, the q-dimensional vector of random effects, and $\mathscr{Y}$, the n-dimensional response vector. In a linear mixed model the unconditional distribution of $\mathscr{B}$ and the conditional distribution, $(\mathscr{Y}|\mathscr{B}=\mathbf{b})$, are both multivariate Gaussian (or "normal") distributions,

$$\begin{aligned}(\mathscr{Y}|\mathscr{B}=\mathbf{b}) &\sim \mathscr{N}(\mathbf{X}\beta+\mathbf{Z}\mathbf{b},\sigma^2\mathbf{I})\\ \mathscr{B} &\sim \mathscr{N}(\mathbf{0},\Sigma_\theta).\end{aligned} \tag{1.1}$$

The *conditional mean* of $\mathscr{Y}$, given $\mathscr{B}=\mathbf{b}$, is the *linear predictor*, $\mathbf{X}\beta+\mathbf{Z}\mathbf{b}$, which depends on the p-dimensional *fixed-effects parameter*, β, and on $\mathbf{b}$. The *model matrices*, $\mathbf{X}$ and $\mathbf{Z}$, of dimension $n\times p$ and $n\times q$, respectively, are determined from the formula for the model and the values of covariates. Although the matrix $\mathbf{Z}$ can be large (i.e. both n and q can be large), it is sparse (i.e. most of the elements in the matrix are zero).

The *relative covariance factor*, Λ_θ is a $q \times q$ matrix, depending on the *variance-component parameter*, θ, and generating the symmetric $q \times q$ variance-covariance matrix, Σ_θ, according to

$$\Sigma_\theta = \sigma^2 \Lambda_\theta \Lambda_\theta^\mathsf{T}. \tag{1.2}$$

The *spherical random effects*, $\mathscr{U} \sim \mathscr{N}(\mathbf{0}, \sigma^2 \mathbf{I}_q)$, determine $\mathscr{B}$ according to

$$\mathscr{B} = \Lambda_\theta \mathscr{U}. \tag{1.3}$$

The *penalized residual sum of squares* (PRSS),

$$r^2(\theta, \beta, \mathbf{u}) = \left\{ \|\mathbf{y} - \mathbf{X}\beta - \mathbf{Z}\Lambda_\theta \mathbf{u}\|^2 + \|\mathbf{u}\|^2 \right\}, \tag{1.4}$$

is the sum of the residual sum of squares, measuring fidelity of the model to the data, and a penalty on the size of $\mathbf{u}$, measuring the complexity of the model. Minimizing r^2 with respect to $\mathbf{u}$,

$$r^2_{\beta,\theta} = \min_{\mathbf{u}} \left\{ \|\mathbf{y} - \mathbf{X}\beta - \mathbf{Z}\Lambda_\theta \mathbf{u}\|^2 + \|\mathbf{u}\|^2 \right\} \tag{1.5}$$

is a direct (i.e. non-iterative) computation for which we calculate the *sparse Cholesky factor*, $\mathbf{L}_\theta$, which is a lower triangular $q \times q$ matrix satisfying

$$\mathbf{L}_\theta \mathbf{L}_\theta^\mathsf{T} = \Lambda_\theta^\mathsf{T} \mathbf{Z}^\mathsf{T} \mathbf{Z} \Lambda_\theta + \mathbf{I}_q. \tag{1.6}$$

where $\mathbf{I}_q$ is the $q \times q$ *identity matrix*.

The *deviance* (negative twice the log-likelihood) of the parameters, given the data, $\mathbf{y}$, is

$$d(\theta, \beta, \sigma | \mathbf{y}) = n \log(2\pi\sigma^2) + \log(|\mathbf{L}_\theta|^2) + \frac{r^2_{\beta,\theta}}{\sigma^2}. \tag{1.7}$$

where $|\mathbf{L}_\theta|$ denotes the *determinant* of $\mathbf{L}_\theta$. Because $\mathbf{L}_\theta$ is triangular, its determinant is the product of its diagonal elements.

Because the conditional mean, μ, is a linear function of β and $\mathbf{u}$, minimization of the PRSS with respect to both β and $\mathbf{u}$ to produce

$$r^2_\theta = \min_{\beta,\mathbf{u}} \left\{ \|\mathbf{y} - \mathbf{X}\beta - \mathbf{Z}\Lambda_\theta \mathbf{u}\|^2 + \|\mathbf{u}\|^2 \right\} \tag{1.8}$$

is also a direct calculation. The values of $\mathbf{u}$ and β that provide this minimum are called, respectively, the *conditional mode*, $\tilde{\mathbf{u}}_\theta$, of the spherical random effects and the conditional estimate, $\widehat{\beta}_\theta$, of the fixed effects. At the conditional estimate of the fixed effects the deviance is

$$d(\theta, \widehat{\beta}_\theta, \sigma | \mathbf{y}) = n \log(2\pi\sigma^2) + \log(|\mathbf{L}_\theta|^2) + \frac{r^2_\theta}{\sigma^2}. \tag{1.9}$$

Minimizing this expression with respect to σ^2 produces the conditional estimate

$$\widehat{\sigma^2}_\theta = \frac{r_\theta^2}{n} \tag{1.10}$$

which provides the *profiled deviance*,

$$\tilde{d}(\theta|\mathbf{y}) = d(\theta, \widehat{\beta}_\theta, \widehat{\sigma}_\theta|\mathbf{y}) = \log(|\mathbf{L}_\theta|^2) + n\left(1 + \log\left(\frac{2\pi r_\theta^2}{n}\right)\right), \tag{1.11}$$

a function of θ alone.

The *maximum likelihood estimate* (MLE) of θ, written $\widehat{\theta}$, is the value that minimizes the profiled deviance (1.11). We determine this value by numerical optimization. In the process of evaluating $\tilde{d}(\widehat{\theta}|\mathbf{y})$ we determine $\widehat{\beta}$, $\tilde{\mathbf{u}}_{\widehat{\theta}}$ and $r^2_{\widehat{\theta}}$, from which we can evaluate $\widehat{\sigma} = \sqrt{r^2_{\widehat{\theta}}/n}$.

The elements of the conditional mode of $\mathscr{B}$, evaluated at the parameter estimates,

$$\tilde{b}_{\widehat{\theta}} = \Lambda_{\widehat{\theta}} \tilde{u}_{\widehat{\theta}} \tag{1.12}$$

are sometimes called the *best linear unbiased predictors* or BLUPs of the random effects. Although it has an appealing acronym, I don't find the term particularly instructive (what is a "linear unbiased predictor" and in what sense are these the "best"?) and prefer the term "conditional mode", which is explained in section 1.6.

1.4.2 Matrices and Vectors in the Fitted Model Object

The optional argument, `verbose = TRUE`, in a call to `lmer` produces output showing the progress of the iterative optimization of $\tilde{d}(\theta|\mathbf{y})$.

```
> fm1ML <- lmer(Yield ~ 1|Batch, Dyestuff, REML = FALSE, verbose = TRUE)

  0:     327.76702:  1.00000
  1:     327.35312: 0.807151
  2:     327.33414: 0.725317
  3:     327.32711: 0.754925
  4:     327.32706: 0.752678
  5:     327.32706: 0.752578
  6:     327.32706: 0.752581
```

The algorithm converges in 6 iterations to a profiled deviance of 327.32706 at θ =0.752581.

The actual values of many of the matrices and vectors defined above are available in the *environment* of the fitted model object, accessed with the `env` function. For example, $\Lambda_{\widehat{\theta}}$ is

```
> env(fm1ML)$Lambda
```

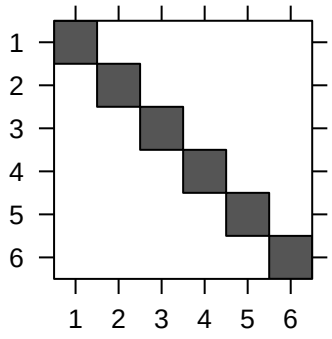


Fig. 1.3 Image of the relative covariance factor, $\Lambda_{\widehat{\theta}}$ for model `fm1ML`. The non-zero elements are shown as darkened squares. The zero elements are blank.

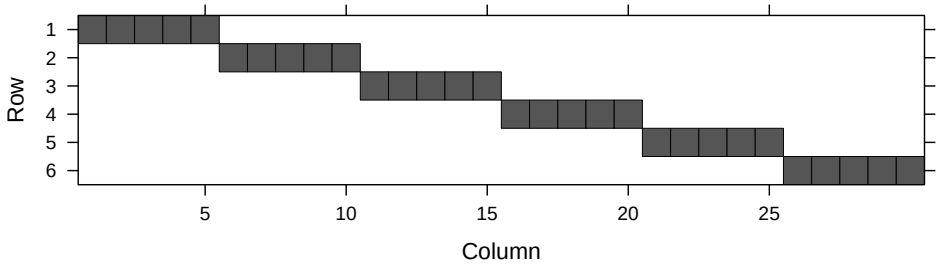


Fig. 1.4 Image of the transpose of the random-effects model matrix, $\mathbf{Z}$, for model `fm1`. The non-zero elements, which are all unity, are shown as darkened squares. The zero elements are blank.

```
6 x 6 diagonal matrix of class "ddiMatrix"
     [,1]      [,2]      [,3]      [,4]      [,5]      [,6]
[1,] 0.7525806         .         .         .         .         .
[2,]         . 0.7525806         .         .         .         .
[3,]         .         . 0.7525806         .         .         .
[4,]         .         .         . 0.7525806         .         .
[5,]         .         .         .         . 0.7525806         .
[6,]         .         .         .         .         . 0.7525806
```

Often we will show the structure of sparse matrices as an image (Fig. 1.3). Especially for large sparse matrices, the image conveys the structure more compactly than does the printed representation.

In this simple model $\Lambda = \theta\mathbf{I}_6$ is a multiple of the identity matrix and the 30×6 model matrix $\mathbf{Z}$, whose transpose is shown in Fig. 1.4, consists of the indicator columns for `Batch`. Because the data are balanced with respect to `Batch`, the Cholesky factor, $\mathbf{L}$ is also a multiple of the identity (you can check this with `image(env(fm1ML)$L)`. The vectors $\mathbf{u}$ and $\mathbf{b}$ and the matrix $\mathbf{X}$ have the same names in `env(fm1ML)`. The vector β is called `fixef`.

1.5 Assessing the Variability of the Parameter Estimates

In this section we show how to create a *profile deviance* object from a fitted linear mixed model and how to use this object to evaluate confidence intervals on the parameters. We also discuss the construction and interpretation of *profile zeta* plots for the parameters and *profile pairs* plots for parameter pairs.

1.5.1 Confidence Intervals on the Parameters

The mixed-effects model fit as `fm1` or `fm1ML` has three parameters for which we obtained estimates. These parameters are σ_1, the standard deviation of the random effects, σ, the standard deviation of the residual or "per-observation" noise term and β_0, the fixed-effects parameter that is labeled as `(Intercept)`.

The `profile` function systematically varies the parameters in a model, assessing the best possible fit that can be obtained with one parameter fixed at a specific value and comparing this fit to the *globally optimal fit*, which is the original model fit that allowed all the parameters to vary. The models are compared according to the change in the deviance, which is the *likelihood ratio test* (LRT) statistic. We apply a *signed square root* transformation to this statistic and plot the resulting function, called ζ, versus the parameter value. A ζ value can be compared to the quantiles of the *standard normal distribution*, $\mathcal{Z} \sim \mathcal{N}(0,1)$. For example, a 95% profile deviance confidence interval on the parameter consists of the values for which $-1.960 < \zeta < 1.960$.

Because the process of profiling a fitted model, which involves re-fitting the model many times, can be computationally intensive, one should exercise caution with complex models fit to very large data sets. Because the statistic of interest is a likelihood ratio, the model is re-fit according to the maximum likelihood criterion, even if the original fit is a REML fit. Thus, there is a slight advantage in starting with an ML fit.

```
> pr1 <- profile(fm1ML)
```

Plots of ζ versus the parameter being profiled (Fig. 1.5) are obtained with

```
> xyplot(pr1, aspect = 1.3)
```

We will refer to such plots as *profile zeta* plots. I usually adjust the aspect ratio of the panels in profile zeta plots to, say, `aspect = 1.3` and frequently set the layout so the panels form a single row (`layout = c(3,1)`, in this case).

The vertical lines in the panels delimit the 50%, 80%, 90%, 95% and 99% confidence intervals, when these intervals can be calculated. Numerical values of the endpoints are returned by the `confint` extractor.

```
> confint(pr1)
```

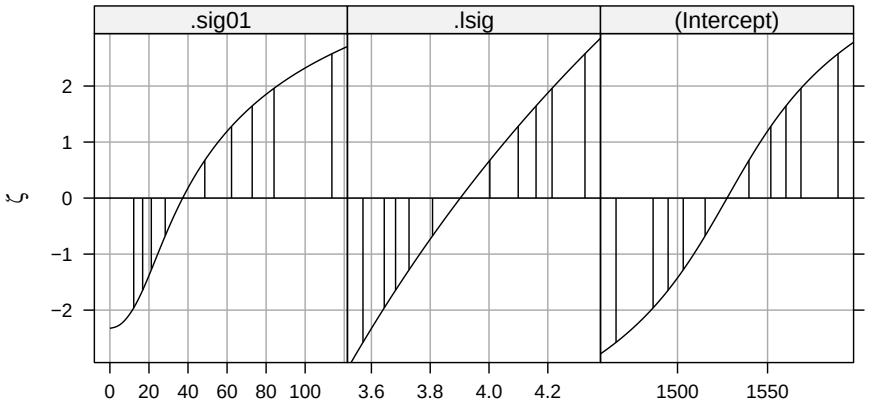


Fig. 1.5 Signed square root, ζ, of the likelihood ratio test statistic for each of the parameters in model `fm1ML`. The vertical lines are the endpoints of 50%, 80%, 90%, 95% and 99% confidence intervals derived from this test statistic.

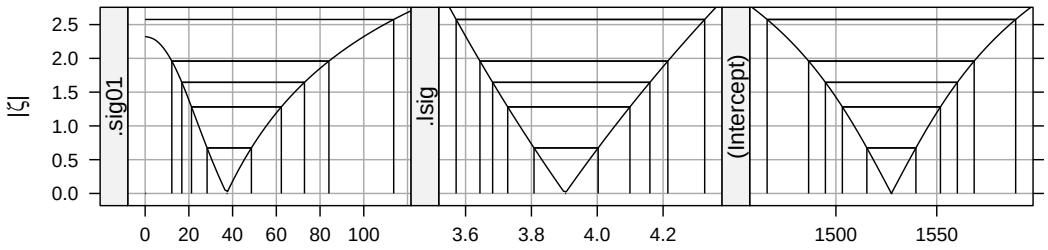


Fig. 1.6 Profiled deviance, on the scale $|\zeta|$, the square root of the change in the deviance, for each of the parameters in model `fm1ML`. The intervals shown are 50%, 80%, 90%, 95% and 99% confidence intervals based on the profile likelihood.

```
                  2.5 %      97.5 %
.sig01        12.197461   84.063361
.lsig          3.643624    4.214461
(Intercept) 1486.451506 1568.548494
```

By default the 95% confidence interval is returned. The optional argument, `level`, is used to obtain other confidence levels.

```
> confint(pr1, level = 0.99)
```

```
                  0.5 %      99.5 %
.sig01               NA  113.690280
.lsig          3.571290    4.326337
(Intercept) 1465.872875 1589.127125
```

Notice that the lower bound on the 99% confidence interval for σ_1 is not defined. Also notice that we profile $\log(\sigma)$ instead of σ, the residual standard deviation.

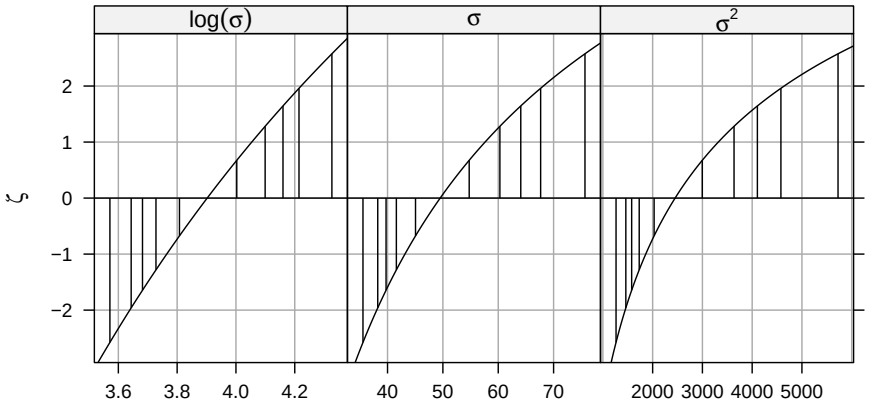


Fig. 1.7 Signed square root, ζ, of the likelihood ratio test statistic as a function of $\log(\sigma)$, of σ and of σ^2. The vertical lines are the endpoints of 50%, 80%, 90%, 95% and 99% confidence intervals.

A plot of $|\zeta|$, the absolute value of ζ, versus the parameter (Fig. 1.6), obtained by adding the optional argument `absVal = TRUE` to the call to `xyplot`, can be more effective for visualizing the confidence intervals.

1.5.2 Interpreting the Profile Zeta Plot

A profile zeta plot, such as Fig. 1.5, shows us the sensitivity of the model fit to changes in the value of particular parameters. Although this is not quite the same as describing the distribution of an estimator, it is a similar idea and we will use some of the terminology from distributions when describing these plots. Essentially we view the patterns in the plots as we would those in a normal probability plot of data values or residuals from a model.

Ideally the profile zeta plot will be close to a straight line over the region of interest, in which case we can perform reliable statistical inference based on the parameter's estimate, its standard error and quantiles of the standard normal distribution. We will describe such as situation as providing a good normal approximation for inference. The common practice of quoting a parameter estimate and its standard error assumes that this is always the case.

In Fig. 1.5 the profile zeta plot for $\log(\sigma)$ is reasonably straight so $\log(\sigma)$ has a good normal approximation. But this does not mean that there is a good normal approximation for σ^2 or even for σ. As shown in Fig. 1.7 the profile zeta plot for $\log(\sigma)$ is slightly skewed, that for σ is moderately skewed and the profile zeta plot for σ^2 is highly skewed. Deviance-based confidence intervals on σ^2 are quite asymmetric, of the form "estimate minus a little, plus a lot".

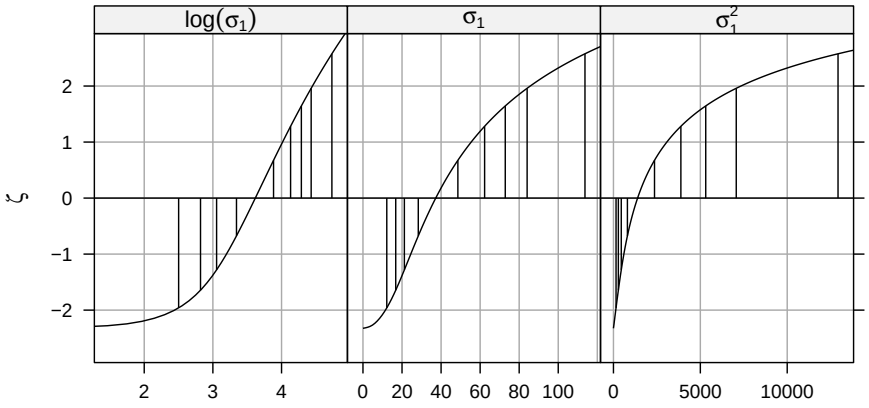


Fig. 1.8 Signed square root, ζ, of the likelihood ratio test statistic as a function of $\log(\sigma_1)$, of σ_1 and of σ_1^2. The vertical lines are the endpoints of 50%, 80%, 90%, 95% and 99% confidence intervals.

This should not come as a surprise to anyone who learned in an introductory statistics course that, given a random sample of data assumed to come from a Gaussian distribution, we use a χ^2 distribution, which can be quite skewed, to form a confidence interval on σ^2. Yet somehow there is a widespread belief that the distribution of variance estimators in much more complex situations should be well approximated by a normal distribution. It is nonsensical to believe that. In most cases summarizing the precision of a variance component estimate by giving an approximate standard error is woefully inadequate.

The pattern in the profile plot for β_0 is sigmoidal (i.e. an elongated "S"-shape). The pattern is symmetric about the estimate but curved in such a way that the profile-based confidence intervals are wider than those based on a normal approximation. We characterize this pattern as symmetric but over-dispersed (relative to a normal distribution). Again, this pattern is not unexpected. Estimators of the coefficients in a linear model without random effects have a distribution which is a scaled Student's T distribution. That is, they follow a symmetric distribution that is over-dispersed relative to the normal.

The pattern in the profile zeta plot for σ_1 is more complex. Fig. 1.8 shows the profile zeta plot on the scale of $\log(\sigma_1)$, σ_1 and σ_1^2. Notice that the profile zeta plot for $\log(\sigma_1)$ is very close to linear to the right of the estimate but flattens out on the left. That is, σ_1 behaves like σ in that its profile zeta plot is more-or-less a straight line on the logarithmic scale, except when σ_1 is close to zero. The model loses sensitivity to values of σ_1 that are close to zero. If, as in this case, zero is within the "region of interest" then we should expect that the profile zeta plot will flatten out on the left hand side.

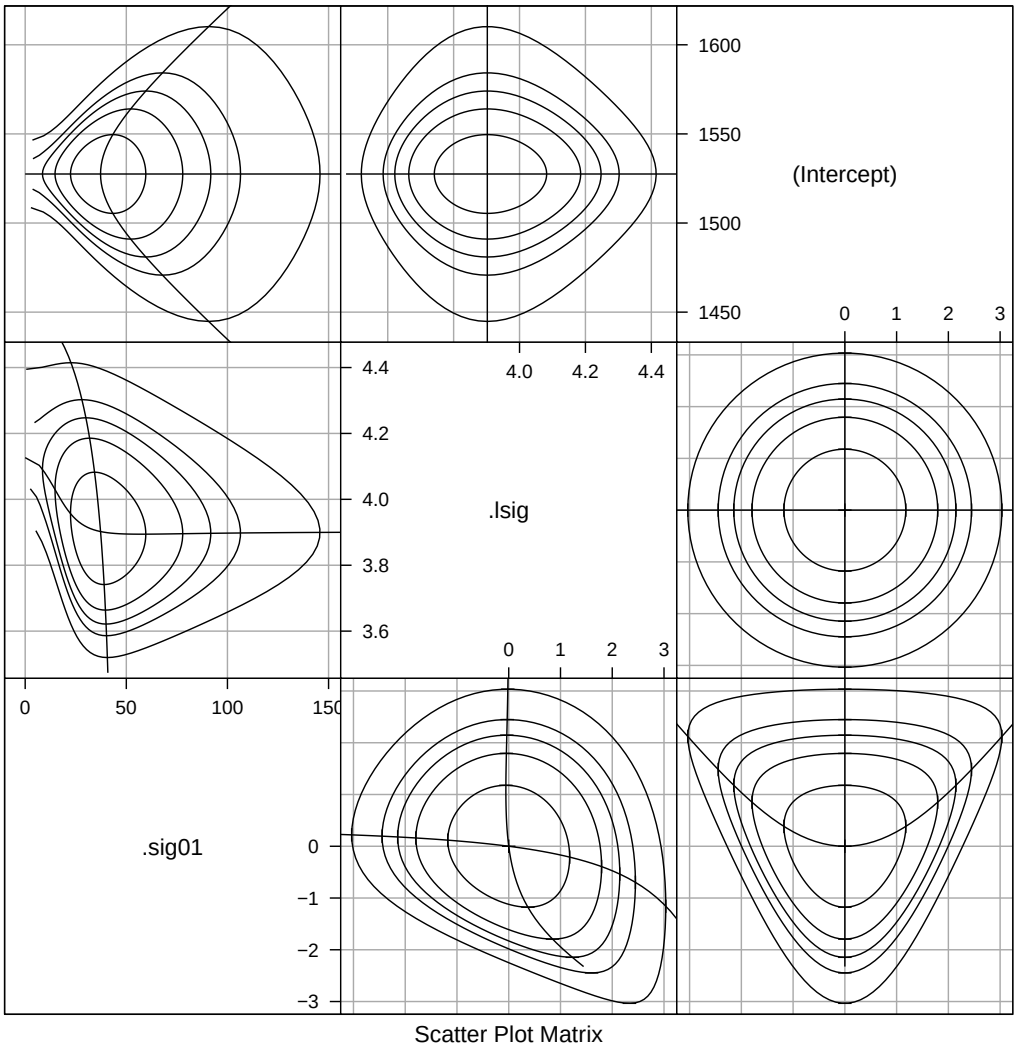


Fig. 1.9 Profile pairs plot for the parameters in model `fm1`. The contour lines correspond to two-dimensional 50%, 80%, 90%, 95% and 99% marginal confidence regions based on the likelihood ratio. Panels below the diagonal represent the (ζ_i, ζ_j) parameters; those above the diagonal represent the original parameters.

1.5.3 Profile Pairs Plots

A profiled deviance object, such as `pr1`, not only provides information on the sensitivity of the model fit to changes in parameters, it also tells us how the parameters influence each other. When we re-fit the model subject to a constraint such as, say, $\sigma_1 = 60$, we obtain the conditional estimates for the other parameters — σ and β_0 in this case. The conditional estimate of, say, σ as a function of σ_1 is called the *profile trace* of σ on σ_1. Plotting such traces provides valuable information on how the parameters in the model are influenced by each other.

The *profile pairs* plot, obtained as

```
> splom(pr1)
```

and shown in Fig. 1.9 shows the profile traces along with interpolated contours of the two-dimensional profiled deviance function. The contours are chosen to correspond to the two-dimensional marginal confidence regions at particular confidence levels.

Because this plot may be rather confusing at first we will explain what is shown in each panel. To make it easier to refer to panels we assign them (x, y) coordinates, as in a Cartesian coordinate system. The columns are numbered 1 to 3 from left to right and the rows are numbered 1 to 3 from bottom to top. Note that the rows are numbered from the bottom to the top, like the y-axis of a graph, not from top to bottom, like a matrix.

The diagonal panels show the ordering of the parameters: σ_1 first, then $\log(\sigma)$ then β_0. Panels above the diagonal are in the original scale of the parameters. That is, the top-left panel, which is the $(1,3)$ position, has σ_1 on the horizontal axis and β_0 on the vertical axis.

In addition to the contour lines in this panel, there are two other lines, which are the profile traces of σ_1 on β_0 and of β_0 on σ_1. The profile trace of β_0 on σ_1 is a straight horizontal line, indicating that the conditional estimate of β_0, given a value of σ_1, is constant. Again, this is a consequence of the simple model form and the balanced data set. The other line in this panel, which is the profile trace of σ_1 on β_0, is curved. That is, the conditional estimate of σ_1 given β_0 depends on β_0. As β_0 moves away from the estimate, $\widehat{\beta}_0$, in either direction, the conditional estimate of σ_1 increases.

We will refer to the two traces on a panel as the "horizontal trace" and "vertical trace". They are not always perfectly horizontal and vertical lines but the meaning should be clear from the panel because one trace will always be more horizontal and the other will be more vertical. The one that is more horizontal is the trace of the parameter on the y axis as a function of the parameter on the horizontal axis, and vice versa.

The contours shown on the panel are interpolated from the profile zeta function and the profile traces, in the manner described in Bates and Watts [1988, Chapter 6]. One characteristic of a profile trace, which we can verify visually in this panel, is that the tangent to a contour must be vertical where it intersects the horizontal trace and horizontal where it intersects the vertical trace.

The $(2,3)$ panel shows β_0 versus $\log(\sigma)$. In this case the traces actually are horizontal and vertical straight lines. That is, the conditional estimate of β_0 doesn't depend on $\log(\sigma)$ and the conditional estimate of $\log(\sigma)$ doesn't depend on β_0. Even in this case, however, the contour lines are not concentric ellipses, because the deviance is not perfectly quadratic in these parameters. That is, the zeta functions, $\zeta(\beta_0)$ and $\zeta(\log(\sigma))$, are not linear.

The $(1,2)$ panel, showing $\log(\sigma)$ versus σ_1 shows distortion along both axes and nonlinear patterns in both traces. When σ_1 is close to zero the conditional estimate of $\log(\sigma)$ is larger than when σ_1 is large. In other words

small values of σ_1 inflate the estimate of $\log(\sigma)$ because the variability that would be explained by the random effects gets incorporated into the residual noise term.

Panels below the diagonal are on the ζ scale, which is why the axes on each of these panels span the same range, approximately -3 to $+3$, and the profile traces always cross at the origin. Thus the $(3,1)$ panel shows $\zeta(\sigma_1)$ on the vertical axis versus $\zeta(\beta_0)$ on the horizontal. These panels allow us to see distortions from an elliptical shape due to nonlinearity of the traces, separately from the one-dimensional distortions caused by a poor choice of scale for the parameter. The ζ scales provide, in some sense, the best possible set of single-parameter transformations for assessing the contours. On the ζ scales the extent of a contour on the horizontal axis is exactly the same as the extent on the vertical axis and both are centered about zero.

Another way to think of this is that, if we would have profiled σ_1^2 instead of σ_1, we would change all the panels in the first column but the panels on the first row would remain the same.

1.6 Assessing the Random Effects

In section 1.4.1 we mentioned that what are sometimes called the BLUPs (or best linear unbiased estimators) of the random effects, $\mathscr{B}$, are the conditional modes evaluated at the parameter estimates, and that they can be calculated as $\tilde{b}_{\widehat{\theta}} = \Lambda_{\widehat{\theta}}\tilde{u}_{\widehat{\theta}}$.

These values are often considered as some sort of "estimates" of the random effects. It can be helpful to think of them this way but it can also be misleading. As we have stated, the random effects are not, strictly speaking, parameters—they are unobserved random variables. We don't estimate the random effects in the same sense that we estimate parameters. Instead, we consider the conditional distribution of $\mathscr{B}$ given the observed data, $(\mathscr{B}|\mathscr{Y} = \mathbf{y})$.

Because the unconditional distribution, $\mathscr{B} \sim \mathscr{N}(\mathbf{0}, \Sigma_\theta)$ is continuous, the conditional distribution, $(\mathscr{B}|\mathscr{Y} = \mathbf{y})$ will also be continuous. In general, the mode of a probability density is the point of maximum density, so the phrase "conditional mode" refers to the point at which this conditional density is maximized. Because this definition relates to the probability model, the values of the parameters are assumed to be known. In practice, of course, we don't know the values of the parameters (if we did there would be no purpose in forming the parameter estimates), so we use the estimated values of the parameters to evaluate the conditional modes.

Those who are familiar with the multivariate Gaussian distribution may recognize that, because both $\mathscr{B}$ and $(\mathscr{Y}|\mathscr{B} = \mathbf{b})$ are multivariate Gaussian, $(\mathscr{B}|\mathscr{Y} = \mathbf{y})$ will also be multivariate Gaussian and the conditional mode will also be the conditional mean of $\mathscr{B}$, given $\mathscr{Y} = \mathbf{y}$. This is the case for a linear

mixed model but it does not carry over to other forms of mixed models. In the general case all we can say about $\tilde{\mathbf{u}}$ or $\tilde{\mathbf{b}}$ is that they maximize a conditional density, which is why we use the term "conditional mode" to describe these values. We will only use the term "conditional mean" and the symbol, μ, in reference to $\mathrm{E}(\mathscr{Y}|\mathscr{B}=\mathbf{b})$, which is the conditional mean of $\mathscr{Y}$ given $\mathscr{B}$, and an important part of the formulation of all types of mixed-effects models.

The `ranef` extractor returns the conditional modes.

```
> ranef(fm1ML)

$Batch
  (Intercept)
A  -16.628221
B    0.369516
C   26.974670
D  -21.801445
E   53.579824
F  -42.494343
```

Applying `str` to the result of `ranef`

```
> str(ranef(fm1ML))

List of 1
 $ Batch:'data.frame':        6 obs. of  1 variable:
  ..$ (Intercept): num [1:6] -16.628 0.37 26.975 -21.801 53.58 ...
 - attr(*, "class")= chr "ranef.mer"
```

shows that the value is a list of data frames. In this case the list is of length 1 because there is only one random-effects term, `(1|Batch)`, in the model and, hence, only one grouping factor, `Batch`, for the random effects. There is only one column in this data frame because the random-effects term, `(1|Batch)`, is a simple, scalar term.

To make this more explicit, random-effects terms in the model formula are those that contain the vertical bar (`"|"`) character. The `Batch` variable is the grouping factor for the random effects generated by this term. An expression for the grouping factor, usually just the name of a variable, occurs to the right of the vertical bar. If the expression on the left of the vertical bar is `1`, as it is here, we describe the term as a *simple, scalar, random-effects term*. The designation "scalar" means there will be exactly one random effect generated for each level of the grouping factor. A simple, scalar term generates a block of indicator columns — the indicators for the grouping factor — in $\mathbf{Z}$. Because there is only one random-effects term in this model and because that term is a simple, scalar term, the model matrix $\mathbf{Z}$ for this model is the indicator matrix for the levels of `Batch`.

In the next chapter we fit models with multiple simple, scalar terms and, in subsequent chapters, we extend random-effects terms beyond simple, scalar terms. When we have only simple, scalar terms in the model, each term has a unique grouping factor and the elements of the list returned by `ranef` can be considered as associated with terms or with grouping factors. In more

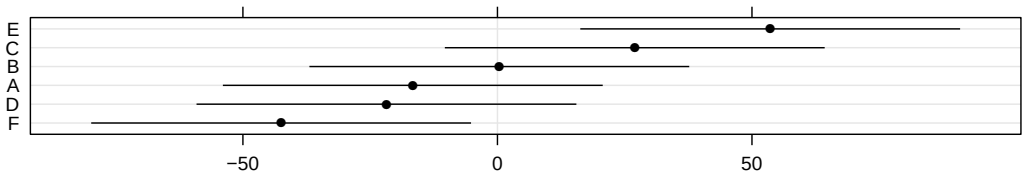
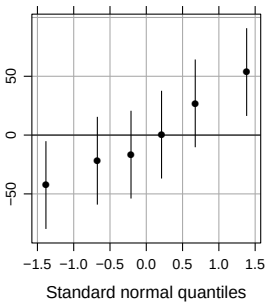


Fig. 1.10 95% prediction intervals on the random effects in `fm1ML`, shown as a dotplot.

Fig. 1.11 95% prediction intervals on the random effects in `fm1ML` versus quantiles of the standard normal distribution.



complex models a particular grouping factor may occur in more than one term, in which case the elements of the list are associated with the grouping factors, not the terms.

Given the data, $\mathbf{y}$, and the parameter estimates, we can evaluate a measure of the dispersion of $(\mathcal{B}|\mathcal{Y} = \mathbf{y})$. In the case of a linear mixed model, this is the conditional standard deviation, from which we can obtain a prediction interval. The `ranef` extractor takes an optional argument, `postVar = TRUE`, which adds these dispersion measures as an attribute of the result. (The name stands for "posterior variance", which is a misnomer that had become established as an argument name before I realized that it wasn't the correct term.)

We can plot these prediction intervals using

```
> dotplot(ranef(fm1ML, postVar = TRUE))
```

(Fig. 1.10), which provides linear spacing of the levels on the y axis, or using

```
> qqmath(ranef(fm1ML, postVar=TRUE))
```

(Fig. 1.11), where the intervals are plotted versus quantiles of the standard normal.

The dotplot is preferred when there are only a few levels of the grouping factor, as in this case. When there are hundreds or thousands of random effects the `qqmath` form is preferred because it focuses attention on the "important few" at the extremes and de-emphasizes the "trivial many" that are close to zero.

1.7 Chapter Summary

A considerable amount of material has been presented in this chapter, especially considering the word "simple" in its title (it's the model that is simple, not the material). A summary may be in order.

A mixed-effects model incorporates fixed-effects parameters and random effects, which are unobserved random variables, $\mathscr{B}$. In a linear mixed model, both the unconditional distribution of $\mathscr{B}$ and the conditional distribution, $(\mathscr{Y}|\mathscr{B}=\mathbf{b})$, are multivariate Gaussian distributions. Furthermore, this conditional distribution is a spherical Gaussian with mean, μ, determined by the linear predictor, $\mathbf{Zb}+\mathbf{X}\beta$. That is,

$$(\mathscr{Y}|\mathscr{B}=\mathbf{b}) \sim \mathscr{N}(\mathbf{Zb}+\mathbf{X}\beta, \sigma^2\mathbf{I}_n).$$

The unconditional distribution of $\mathscr{B}$ has mean $\mathbf{0}$ and a parameterized $q \times q$ variance-covariance matrix, Σ_θ.

In the models we considered in this chapter, Σ_θ, is a simple multiple of the identity matrix, $\mathbf{I}_6$. This matrix is always a multiple of the identity in models with just one random-effects term that is a simple, scalar term. The reason for introducing all the machinery that we did is to allow for more general model specifications.

The maximum likelihood estimates of the parameters are obtained by minimizing the deviance. For linear mixed models we can minimize the profiled deviance, which is a function of θ only, thereby considerably simplifying the optimization problem.

To assess the precision of the parameter estimates, we profile the deviance function with respect to each parameter and apply a signed square root transformation to the likelihood ratio test statistic, producing a profile zeta function for each parameter. These functions provide likelihood-based confidence intervals for the parameters. Profile zeta plots allow us to visually assess the precision of individual parameters. Profile pairs plots allow us to visualize the pairwise dependence of parameter estimates and two-dimensional marginal confidence regions.

Prediction intervals from the conditional distribution of the random effects, given the observed data, allow us to assess the precision of the random effects.

Exercises

These exercises and several others in this book use data sets from the `MEMSS` package for R. You will need to ensure that this package is installed before you can access the data sets.

To load a particular data set, either attach the package

```
> library(MEMSS)
```

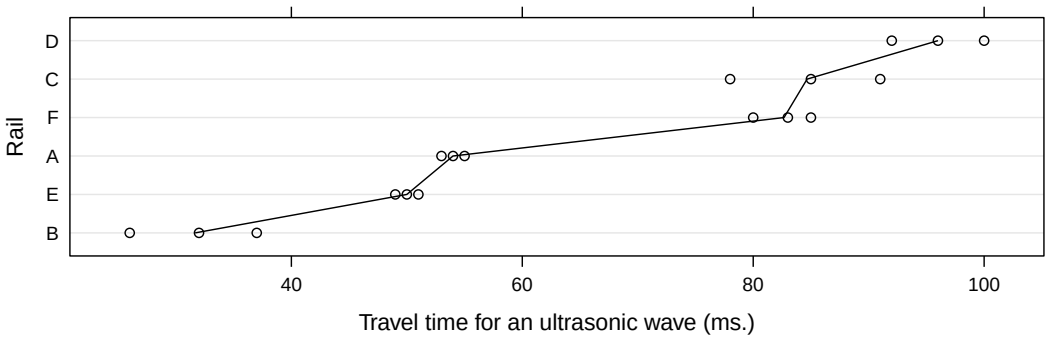


Fig. 1.12 Travel time for an ultrasonic wave test on 6 rails

or load just the one data set

```
> data(Rail, package = "MEMSS")
```

1.1. Check the documentation, the structure (`str`) and a summary of the `Rail` data (Fig. 1.12) from the `MEMSS` package. Note that if you used `data` to access this data set then you must use

```
> help(Rail, package = "MEMSS")
```

to display the documentation for it.

1.2. Fit a model with `travel` as the response and a simple, scalar random-effects term for the variable `Rail`. Use the REML criterion, which is the default. Create a dotplot of the conditional modes of the random effects.

1.3. Refit the model using maximum likelihood. Check the parameter estimates and, in the case of the fixed-effects parameter, its standard error. In what ways have the parameter estimates changed? Which parameter estimates have not changed?

1.4. Profile the fitted model and construct 95% profile-based confidence intervals on the parameters. Is the confidence interval on σ_1 close to being symmetric about the estimate? Is the corresponding interval on $\log(\sigma_1)$ close to being symmetric about its estimate?

1.5. Create the profile zeta plot for this model. For which parameters there good normal approximations?

1.6. Create a profile pairs plot for this model. Does the shape of the deviance contours in this model mirror those in Fig. 1.9?

1.7. Plot the prediction intervals on the random effects from this model. Do any of these prediction intervals contain zero? Consider the relative magnitudes of $\widehat{\sigma_1}$ and $\widehat{\sigma}$ in this model compared to those in model `fm1` for the `Dyestuff` data. Should these ratios of σ_1/σ lead you to expect a different pattern of prediction intervals in this plot than those in Fig. 1.10?

Chapter 2
Models with multiple random-effects terms

The mixed models considered in the previous chapter had only one random-effects term, which was a simple, scalar random-effects term, and a single fixed-effects coefficient. Although such models can be useful, it is with the facility to use multiple random-effects terms and to use random-effects terms beyond a simple, scalar term that we can begin to realize the flexibility and versatility of mixed models.

In this chapter we consider models with multiple simple, scalar random-effects terms, showing examples where the grouping factors for these terms are in completely crossed or nested or partially crossed configurations. For ease of description we will refer to the random effects as being crossed or nested although, strictly speaking, the distinction between nested and non-nested refers to the grouping factors, not the random effects.

2.1 A model With Crossed Random Effects

One of the areas in which the methods in the `lme4` package for R are particularly effective is in fitting models to cross-classified data where several factors have random effects associated with them. For example, in many experiments in psychology the reaction of each of a set of subjects to each of a group of stimuli or items is measured. If the subjects are considered to be a sample from a population of subjects and the items are a sample from a population of items, then it would make sense to associate random effects with both these factors.

In the past it was difficult to fit mixed models with multiple, crossed grouping factors to large, possibly unbalanced, data sets. The methods in the `lme4` package are able to do this. To introduce the methods let us first consider a small, balanced data set with crossed grouping factors.

2.1.1 The `Penicillin` Data

The `Penicillin` data are derived from Table 6.6, p. 144 of Davies and Goldsmith [1972] where they are described as coming from an investigation to

> assess the variability between samples of penicillin by the *B. subtilis* method. In this test method a bulk-innoculated nutrient agar medium is poured into a Petri dish of approximately 90 mm. diameter, known as a plate. When the medium has set, six small hollow cylinders or pots (about 4 mm. in diameter) are cemented onto the surface at equally spaced intervals. A few drops of the penicillin solutions to be compared are placed in the respective cylinders, and the whole plate is placed in an incubator for a given time. Penicillin diffuses from the pots into the agar, and this produces a clear circular zone of inhibition of growth of the organisms, which can be readily measured. The diameter of the zone is related in a known way to the concentration of penicillin in the solution.

As with the `Dyestuff` data, we examine the structure

```
> str(Penicillin)

'data.frame':        144 obs. of  3 variables:
 $ diameter: num  27 23 26 23 23 21 27 23 26 23 ...
 $ plate   : Factor w/ 24 levels "a","b","c","d",..: 1 1 1 1 1 1 2 2 2 2 ...
 $ sample  : Factor w/ 6 levels "A","B","C","D",..: 1 2 3 4 5 6 1 2 3 4 ...
```

and a summary

```
> summary(Penicillin)

    diameter         plate     sample
 Min.   :18.00   a      :  6   A:24
 1st Qu.:22.00   b      :  6   B:24
 Median :23.00   c      :  6   C:24
 Mean   :22.97   d      :  6   D:24
 3rd Qu.:24.00   e      :  6   E:24
 Max.   :27.00   f      :  6   F:24
                 (Other):108
```

of the `Penicillin` data, then plot it (Fig. 2.1).

The variation in the diameter is associated with the plates and with the samples. Because each plate is used for only the six samples shown here we will use random effects for the plate. As in the dyestuff example, we are more interested in the sample-to-sample variability in the penicillin samples than in the potency of a particular sample. Hence we will also use random effects for the sample.

In this experiment each sample is used on each plate. We say that the `sample` and `plate` factors are *crossed*, as opposed to *nested* factors, which we will describe in the next section. By itself, the designation "crossed" just means that the factors are not nested. If we wish to be more specific, we could describe these factors as being *completely crossed*, which means that we have at least one observation for each combination of a level of `sample` and

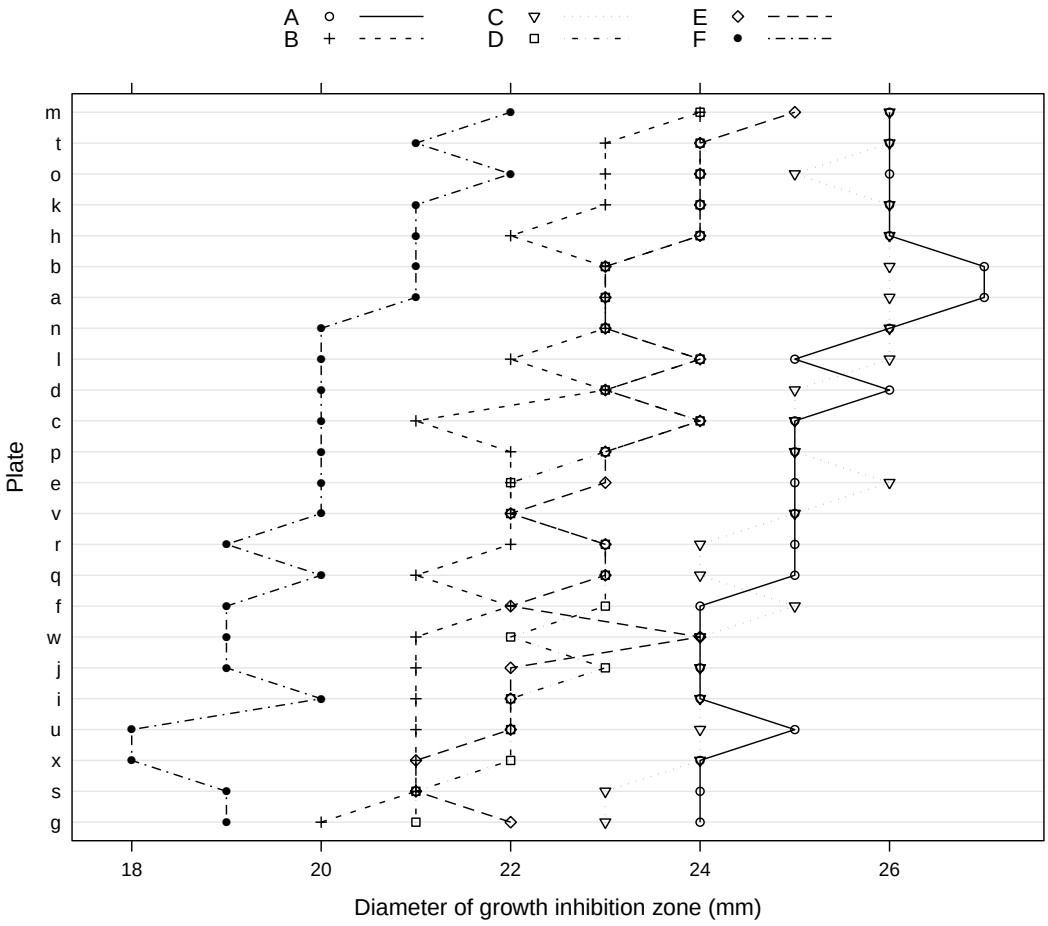


Fig. 2.1 Diameter of the growth inhibition zone (mm) in the *B. subtilis* method of assessing the concentration of penicillin. Each of 6 samples was applied to each of the 24 agar plates. The lines join observations on the same sample.

a level of `plate`. We can see this in Fig. 2.1 and, because there are moderate numbers of levels in these factors, we can check it in a cross-tabulation

```
> xtabs(~ sample + plate, Penicillin)
```

```
      plate
sample a b c d e f g h i j k l m n o p q r s t u v w x
     A 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
     B 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
     C 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
     D 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
     E 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
     F 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
```

Like the `Dyestuff` data, the factors in the `Penicillin` data are balanced. That is, there are exactly the same number of observations on each plate and for each sample and, furthermore, there is the same number of observations on each combination of levels. In this case there is exactly one observation for

each combination of sample and plate. We would describe the configuration of these two factors as an unreplicated, completely balanced, crossed design.

In general, balance is a desirable but precarious property of a data set. We may be able to impose balance in a designed experiment but we typically cannot expect that data from an observation study will be balanced. Also, as anyone who analyzes real data soon finds out, expecting that balance in the design of an experiment will produce a balanced data set is contrary to "Murphy's Law". That's why statisticians allow for missing data. Even when we apply each of the six samples to each of the 24 plates, something could go wrong for one of the samples on one of the plates, leaving us without a measurement for that combination of levels and thus an unbalanced data set.

2.1.2 A Model for the `Penicillin` *Data*

A model incorporating random effects for both the `plate` and the `sample` is straightforward to specify — we include simple, scalar random effects terms for both these factors.

```
> (fm2 <- lmer(diameter ~ 1 + (1|plate) + (1|sample), Penicillin))
```

```
Linear mixed model fit by REML
Formula: diameter ~ 1 + (1 | plate) + (1 | sample)
   Data: Penicillin
 REML
330.9

Random effects:
 Groups   Name        Variance Std.Dev.
 plate    (Intercept) 0.71691  0.84671
 sample   (Intercept) 3.73097  1.93157
 Residual             0.30241  0.54992
Number of obs: 144, groups: plate, 24; sample, 6

Fixed effects:
            Estimate Std. Error t value
(Intercept)  22.9722     0.8086   28.41
```

This model display indicates that the sample-to-sample variability has the greatest contribution, then plate-to-plate variability and finally the "residual" variability that cannot be attributed to either the sample or the plate. These conclusions are consistent with what we see in the `Penicillin` data plot (Fig. 2.1).

The prediction intervals on the random effects (Fig. 2.10) confirm that the conditional distribution of the random effects for `sample` has much less variability than does the conditional distribution of the random effects for `sample`.

In chapter 1 we saw that a model with a single, simple, scalar random-effects term generated a random-effects model matrix, $\mathbf{Z}$, that is the matrix of

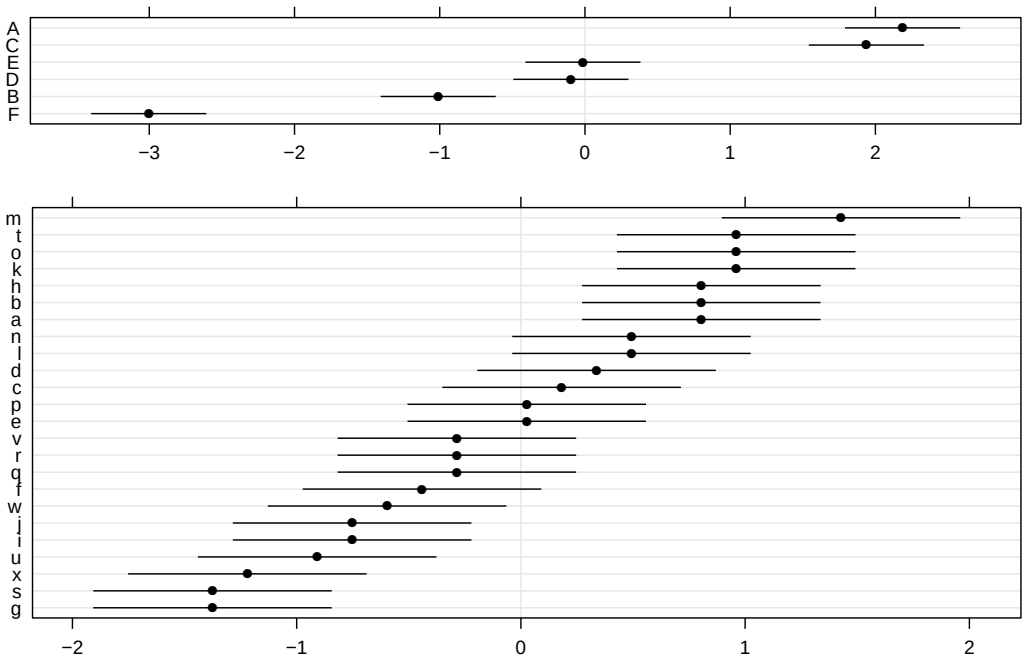


Fig. 2.2 95% prediction intervals on the random effects for model `fm2` fit to the `Penicillin` data.

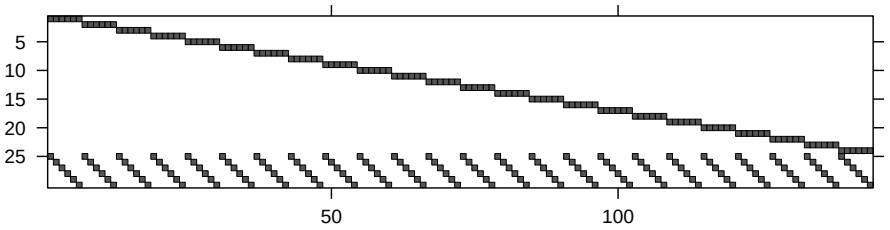


Fig. 2.3 Image of the transpose of the random-effects model matrix, $\mathbf{Z}$, for model `fm2`. The non-zero elements, which are all unity, are shown as darkened squares. The zero elements are blank.

indicators of the levels of the grouping factor. When we have multiple, simple, scalar random-effects terms, as in model `fm2`, each term generates a matrix of indicator columns and all the sets of indicator columns are concatenated to form the model matrix $\mathbf{Z}$. The transpose of this matrix contains rows of indicators for each factor, as shown in Fig. 2.3.

The relative covariance factor (Fig. 2.4, left panel) is no longer a multiple of the identity. It is now block diagonal, with two blocks, one of size 24 and one of size 6, each of which is a multiple of the identity. The diagonal elements in each block are θ_1 and θ_2. The numeric values of these parameters can be obtained as

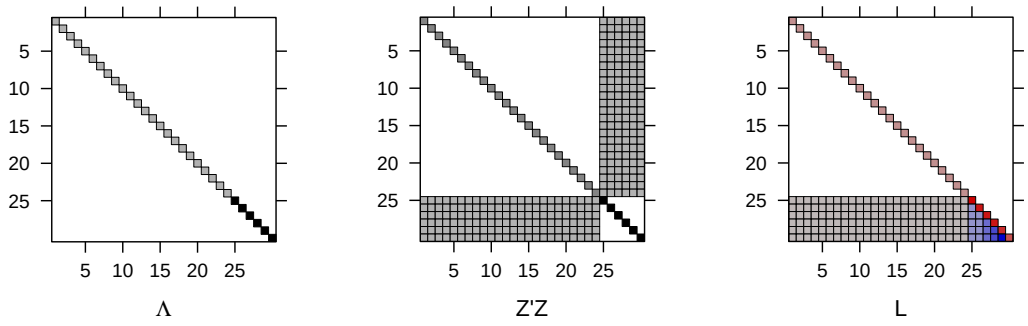


Fig. 2.4 Images of the relative covariance factor, Λ, the cross-product of the random-effects model matrix, $\mathbf{Z}^\mathsf{T}\mathbf{Z}$, and the sparse Cholesky factor, $\mathbf{L}$, for model `fm2`.

```
> env(fm2)$theta

[1] 1.539683 3.512443
```

The first parameter is the relative standard deviation of the random effects for `plate`, which has the value (0.84671/0.54992) at convergence, and the second is the relative standard deviation of the random effects for `sample` (1.93157/0.54992).

Because Λ_θ is diagonal, the pattern of non-zeros in $\Lambda_\theta^\mathsf{T}\mathbf{Z}^\mathsf{T}\mathbf{Z}\Lambda_\theta + \mathbf{I}$ will be the same as that in $\mathbf{Z}^\mathsf{T}\mathbf{Z}$, shown in the middle panel of Fig. 2.4. The sparse Cholesky factor, $\mathbf{L}$, shown in the right panel is lower triangular and has non-zero elements in the lower right hand corner in positions where $\mathbf{Z}^\mathsf{T}\mathbf{Z}$ has systematic zeros. We say that "fill-in" has occurred when forming the sparse Cholesky decomposition. In this case there is a relatively minor amount of fill but in other cases there can be a substantial amount of fill and we shall take precautions so as to reduce this, because fill-in adds to the computational effort in determining the MLEs or the REML estimates.

A profile zeta plot (Fig. 2.5) for the parameters in model `fm2` leads to conclusions similar to those from Fig. 1.5 for model `fm1ML` in the previous chapter. The fixed-effect parameter, β_0, for the `(Intercept)` term has symmetric intervals and is over-dispersed relative to the normal distribution. The logarithm of σ has a good normal approximation but the standard deviations of the random effects, σ_1 and σ_2, are skewed. The skewness for σ_2 is worse than that for σ_1, because the estimate of σ_2 is less precise than that of σ_1, in both absolute and relative senses. For an absolute comparison we compare the widths of the confidence intervals for these parameters.

```
> confint(pr2)

                2.5 %     97.5 %
.sig01      0.6335658  1.1821040
```

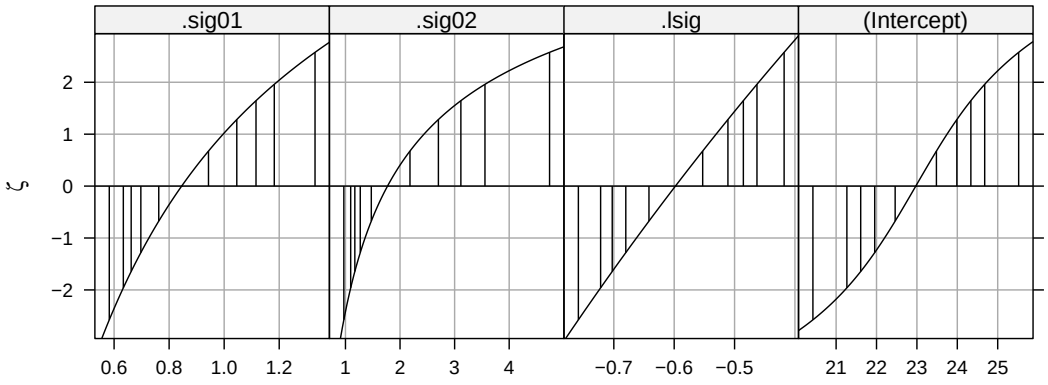


Fig. 2.5 Profile zeta plot of the parameters in model `fm2`.

```
.sig02       1.0957822  3.5563194
.lsig       -0.7218645 -0.4629033
(Intercept) 21.2666274 24.6778176
```

In a relative comparison we examine the ratio of the endpoints of the interval divided by the estimate.

```
> confint(pr2)[1:2,]/c(0.8455722, 1.770648)

           2.5 %   97.5 %
.sig01 0.7492746 1.397993
.sig02 0.6188594 2.008485
```

The lack of precision in the estimate of σ_2 is a consequence of only having 6 distinct levels of the `sample` factor. The `plate` factor, on the other hand, has 24 distinct levels. In general it is more difficult to estimate a measure of spread, such as the standard deviation, than to estimate a measure of location, such as a mean, especially when the number of levels of the factor is small. Six levels are about the minimum number required for obtaining sensible estimates of standard deviations for simple, scalar random effects terms.

The profile pairs plot (Fig. 2.6) shows patterns similar to those in Fig. 1.9 for pairs of parameters in model `fm1` fit to the `Dyestuff` data. On the ζ scale (panels below the diagonal) the profile traces are nearly straight and orthogonal with the exception of the trace of $\zeta(\sigma_2)$ on $\zeta(\beta_0)$ (the horizontal trace for the panel in the $(4,2)$ position. The pattern of this trace is similar to the pattern of the trace of $\zeta(\sigma_1)$ on $\zeta(\beta_0)$ in Fig. 1.9. Moving β_0 from its estimate, $\widehat{\beta}_0$ in either direction will increase the residual sum of squares. The increase in the residual variability is reflected in an increase of one or more of the dispersion parameters. The balanced experimental design results in a fixed estimate of σ and the extra apparent variability must be incorporated into σ_1 or σ_2.

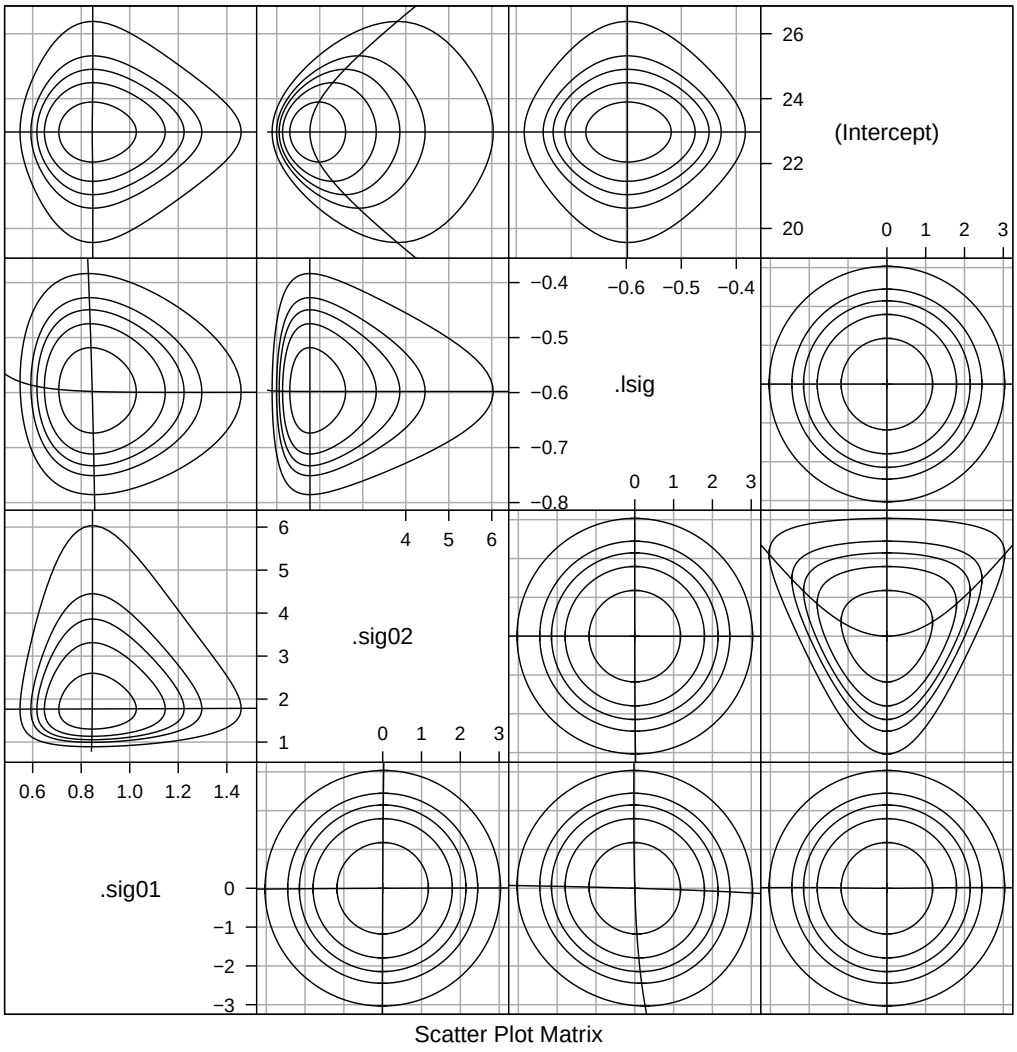


Fig. 2.6 Profile pairs plot for the parameters in model `fm2` fit to the `Penicillin` data.

Contours in panels of parameter pairs on the original scales (i.e. panels above the diagonal) can show considerable distortion from the ideal elliptical shape. For example, contours in the σ_2 versus σ_1 panel (the $(1,2)$ position) and the $\log(\sigma)$ versus σ_2 panel (in the $(2,3)$ position) are dramatically non-elliptical. However, the distortion of the contours is not due to these parameter estimates depending on each other. It is almost entirely due to the choice of scale for σ_1 and σ_2. When we plot the contours on the scale of $\log(\sigma_1)$ and $\log(\sigma_2)$ instead (Fig. **??**) they are much closer to the elliptical pattern.

Conversely, if we tried to plot contours on the scale of σ_1^2 and σ_2^2 (not shown), they would be hideously distorted.

2.2 A model With Nested Random Effects

In this section we again consider a simple example, this time fitting a model with *nested* grouping factors for the random effects.

2.2.1 The Pastes *Data*

The third example from Davies and Goldsmith [1972, Table 6.5, p. 138] is described as coming from

> deliveries of a chemical paste product contained in casks where, in addition to sampling and testing errors, there are variations in quality between deliveries ... As a routine, three casks selected at random from each delivery were sampled and the samples were kept for reference. ... Ten of the delivery batches were sampled at random and two analytical tests carried out on each of the 30 samples.

The structure and summary of the `Pastes` data object are

```
> str(Pastes)

'data.frame':        60 obs. of  4 variables:
 $ strength: num  62.8 62.6 60.1 62.3 62.7 63.1 60 61.4 57.5 56.9 ...
 $ batch   : Factor w/ 10 levels "A","B","C","D",..: 1 1 1 1 1 1 2 2 2 2 ...
 $ cask    : Factor w/ 3 levels "a","b","c": 1 1 2 2 3 3 1 1 2 2 ...
 $ sample  : Factor w/ 30 levels "A:a","A:b","A:c",..: 1 1 2 2 3 3 4 4 5 5 ...
```

```
> summary(Pastes)

    strength         batch    cask       sample
 Min.   :54.20   A      : 6   a:20   A:a    : 2
 1st Qu.:57.50   B      : 6   b:20   A:b    : 2
 Median :59.30   C      : 6   c:20   A:c    : 2
 Mean   :60.05   D      : 6          B:a    : 2
 3rd Qu.:62.88   E      : 6          B:b    : 2
 Max.   :66.00   F      : 6          B:c    : 2
                 (Other):24          (Other):48
```

As stated in the description in Davies and Goldsmith [1972], there are 30 samples, three from each of the 10 delivery batches. We have labelled the levels of the `sample` factor with the label of the `batch` factor followed by 'a', 'b' or 'c' to distinguish the three samples taken from that batch. The cross-tabulation produced by the `xtabs` function, using the optional argument `sparse = TRUE`, provides a concise display of the relationship.

```
> xtabs(~ batch + sample, Pastes, drop = TRUE, sparse = TRUE)

10 x 30 sparse Matrix of class "dgCMatrix"

A 2 2 2 . . . . . .
B . . . 2 2 2 . . . .
```

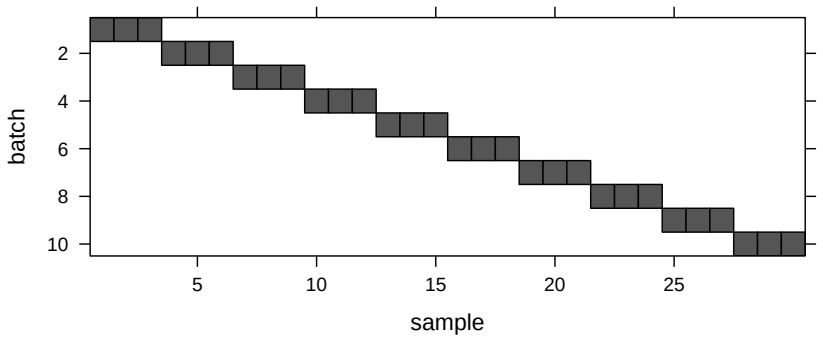


Fig. 2.7 Image of the cross-tabulation of the `batch` and `sample` factors in the `Pastes` data.

```
C . . . . . 2 2 2 . . . . . .
D . . . . . 2 2 2 . . . . . .
E . . . . . 2 2 2 . . . . . .
F . . . . . 2 2 2 . . . . . .
G . . . . . 2 2 2 . . . . . .
H . . . . . 2 2 2 . . . . . .
I . . . . . 2 2 2 . . .
J . . . . . 2 2 2
```

Alternatively, we can use an image (Fig. 2.7) of this cross-tabulation to visualize the structure.

When plotting the `strength` versus `batch` and `sample` in the `Pastes` data we should remember that we have two strength measurements on each of the 30 samples. It is tempting to use the cask designation ('a', 'b' and 'c') to determine, say, the plotting symbol within a `batch`. It would be fine to do this within a batch but the plot would be misleading if we used the same symbol for cask 'a' in different batches. There is no relationship between cask 'a' in batch 'A' and cask 'a' in batch 'B'. The labels 'a', 'b' and 'c' are used only to distinguish the three samples within a batch; they do not have a meaning across batches.

In Fig. 2.8 we plot the two strength measurements on each of the samples within each of the batches and join up the average strength for each sample. The perceptive reader will have noticed that the levels of the factors on the vertical axis in this figure, and in Fig. 1.1 and 2.1, have been reordered according to increasing average response. In all these cases there is no inherent ordering of the levels of the covariate such as `batch` or `plate`. Rather than confuse our interpretation of the plot by determining the vertical displacement of points according to a random ordering, we impose an ordering according to increasing mean response. This allows us to more easily check for structure in the data, including undesirable characteristics like increasing variability of the response with increasing mean level of the response.

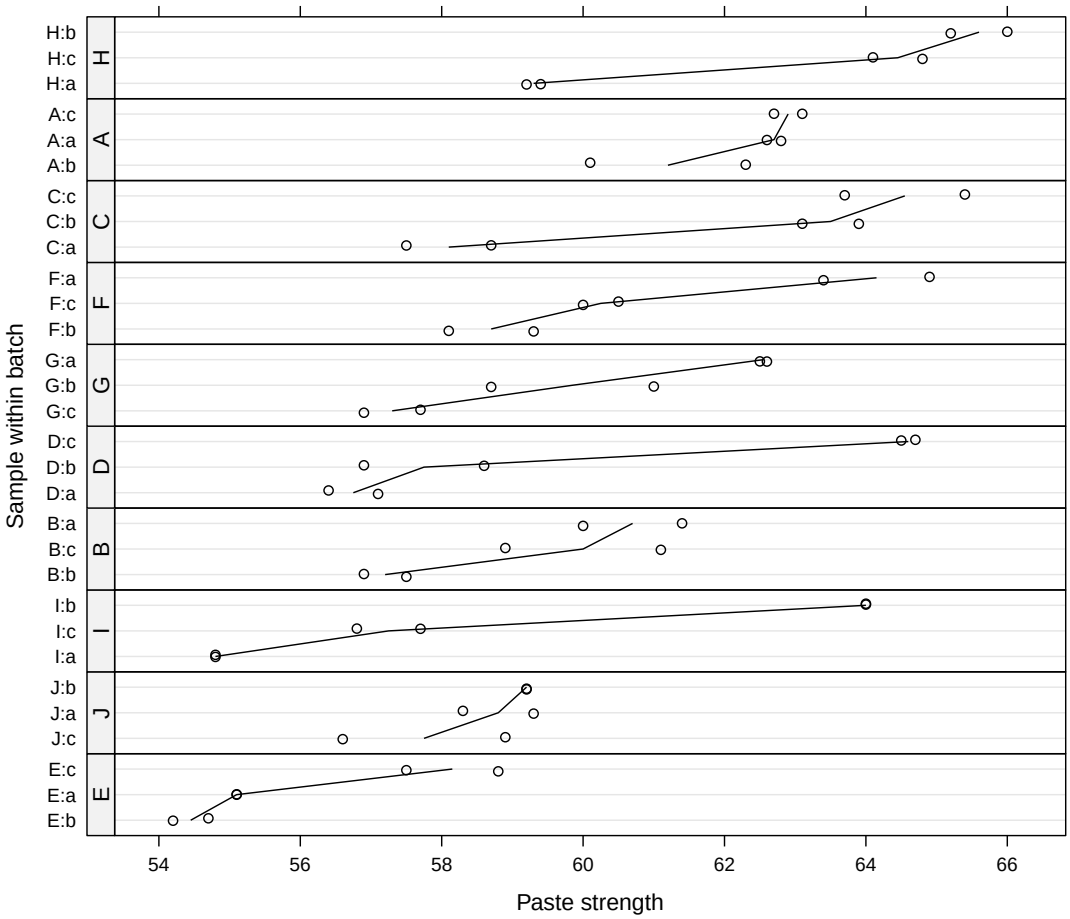


Fig. 2.8 Strength of paste preparations according to the `batch` and the `sample` within the batch. There were two strength measurements on each of the 30 samples; three samples each from 10 batches.

In Fig. 2.8 we order the samples within each batch separately then order the batches according to increasing mean strength.

Figure 2.8 shows considerable variability in strength between samples relative to the variability within samples. There is some indication of variability between batches, in addition to the variability induced by the samples, but not a strong indication of a batch effect. For example, each of batches I and D, with low mean strength relative to the other batches, contained one sample (I:b and D:c, respectively) that had high mean strength relative to the other samples. Also, batches H and C, with comparatively high mean batch strength. contain samples H:a and C:a with comparatively low mean sample strength. In section **??** we will examine the need for incorporating batch-to-batch variability in a statistical model in addition to sample-to-sample variability.

2.2.1.1 Nested Factors

Because each level of `sample` occurs with one and only one level of `batch` we say that `sample` is *nested within* `batch`. Some presentations of mixed-effects models, especially those related to *multilevel modeling* [Rasbash et al., 2000] or *hierarchical linear models* [Raudenbush and Bryk, 2002], leave the impression that one can only define random effects with respect to factors that are nested. This is the origin of the terms "multilevel", referring to multiple, nested levels of variability, and "hierarchical", also invoking the concept of a hierarchy of levels. To be fair, both those references do describe the use of models with random effects associated with non-nested factors, but such models tend to be treated as a special case.

The blurring of mixed-effects models with the concept of multiple, hierarchical levels of variation results in an unwarranted emphasis on "levels" when defining a model and leads to considerable confusion. It is perfectly legitimate to define models having random effects associated with non-nested factors. The reasons for the emphasis on defining random effects with respect to nested factors only are that such cases do occur frequently in practice and that some of the computational methods for estimating the parameters in the models can only be easily applied to nested factors.

This is not the case for the methods used in the `lme4` package. Indeed there is nothing special done for models with random effects for nested factors. When random effects are associated with multiple factors exactly the same computational methods are used whether the factors form a nested sequence or are partially crossed or are completely crossed. A case of a nested sequence of "grouping factors" for the random effects (including the trivial case of only one such factor) is detected but this information does not change the course of the computation. It is available to be used as a diagnostic check. When the user knows that the grouping factors should be nested, she can check if they are indeed nested.

There is, however, one aspect of nested grouping factors that we should emphasize, which is the possibility of a factor that is *implicitly nested* within another factor. Suppose, for example, that the `sample` factor was defined as having three levels instead of 30 with the implicit assumption that `sample` is nested within `batch`. It may seem silly to try to distinguish 30 different batches with only three levels of a factor but, unfortunately, data are frequently organized and presented like this, especially in text books. The `cask` factor in the `Pastes` data is exactly such an implicitly nested factor. If we cross-tabulate `batch` and `cask`

```
> xtabs(~ cask + batch, Pastes)
```

```
    batch
cask A B C D E F G H I J
   a 2 2 2 2 2 2 2 2 2 2
   b 2 2 2 2 2 2 2 2 2 2
   c 2 2 2 2 2 2 2 2 2 2
```

are crossed, not nested. If we know that the cask should be considered as nested within the batch then we should create a new categorical variable giving the batch-cask combination, which is exactly what the `sample` factor is. A simple way to create such a factor is to use the interaction operator, ':', on the factors. It is advisable, but not necessary, to apply `factor` to the result thereby dropping unused levels of the interaction from the set of all possible levels of the factor. (An "unused level" is a combination that does not occur in the data.) A convenient code idiom is

```
> Pastes$sample <- with(Pastes, factor(batch:cask))
```

or

```
> Pastes <- within(Pastes, sample <- factor(batch:cask))
```

In a small data set like `Pastes` we can quickly detect a factor being implicitly nested within another factor and take appropriate action. In a large data set, perhaps hundreds of thousands of test scores for students in thousands of schools from hundreds of school districts, it is not always obvious if school identifiers are unique across the entire data set or just within a district. If you are not sure, the safest thing to do is to create the interaction factor, as shown above, so you can be confident that levels of the district:school interaction do indeed correspond to unique schools.

2.2.2 Fitting a Model With Random-effects for Nested Factors

Fitting a model with simple, scalar random effects for nested factors is done in exactly the same way as fitting a model with random effects for crossed grouping factors. We include random-effects terms for each factor, as in

```
> (fm3 <- lmer(strength ~ 1 + (1|sample) + (1|batch), Pastes, REML=0))

Linear mixed model fit by maximum likelihood
Formula: strength ~ 1 + (1 | sample) + (1 | batch)
   Data: Pastes
 AIC   BIC logLik deviance
 256 264.4   -124      248

Random effects:
 Groups   Name        Variance Std.Dev.
 sample   (Intercept) 8.4337   2.9041
 batch    (Intercept) 1.1992   1.0951
 Residual             0.6780   0.8234
Number of obs: 60, groups: sample, 30; batch, 10

Fixed effects:
            Estimate Std. Error t value
(Intercept)  60.0533     0.6421   93.52
```

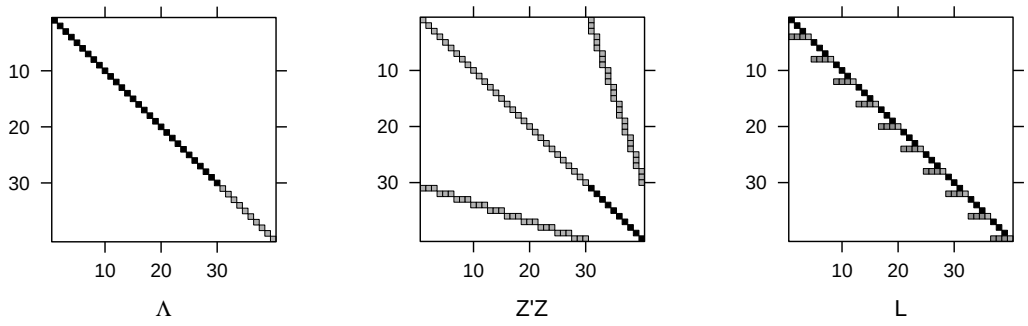


Fig. 2.9 Images of the relative covariance factor, Λ, the cross-product of the random-effects model matrix, $\mathbf{Z}^\mathsf{T}\mathbf{Z}$, and the sparse Cholesky factor, $\mathbf{L}$, for model `fm3`.

Not only is the model specification similar for nested and crossed factors, the internal calculations are performed according to the methods described in Sect. 1.4.1 for each model type. Comparing the patterns in the matrices Λ, $\mathbf{Z}^\mathsf{T}\mathbf{Z}$ and $\mathbf{L}$ for this model (Fig. 2.9) to those in Fig. 2.4 shows that models with nested factors produce simple repeated structures along the diagonal of the sparse Cholesky factor, $\mathbf{L}$, after reordering the random effects (we discuss this reordering later in Sect. **??**). This type of structure has the desirable property that there is no "fill-in" during calculation of the Cholesky factor. In other words, the number of non-zeros in $\mathbf{L}$ is the same as the number of non-zeros in the lower triangle of the matrix being factored, $\Lambda^\mathsf{T}\mathbf{Z}^\mathsf{T}\mathbf{Z}\Lambda + \mathbf{I}$ (which, because Λ is diagonal, has the same structure as $\mathbf{Z}^\mathsf{T}\mathbf{Z}$).

Fill-in of the Cholesky factor is not an important issue when we have a few dozen random effects, as we do here. It is an important issue when we have millions of random effects in complex configurations, as has been the case in some of the models that have been fit using `lmer`.

2.2.3 Assessing the Parameter Estimates for Model `fm3`

The parameter estimates are: $\widehat{\sigma_1}$ =2.904, the standard deviation of the random effects for `sample`; $\widehat{\sigma_2}$ =1.095, the standard deviation of the random effects for `batch`; $\widehat{\sigma}$ =0.823, the standard deviation of the residual noise term; and $\widehat{\beta_0}$ =60.053, the overall mean response, which is labeled `(Intercept)` in these models.

The estimated standard deviation for `sample` is nearly three times as large as that for `batch`, which confirms what we saw in Fig. 2.8. Indeed our con-

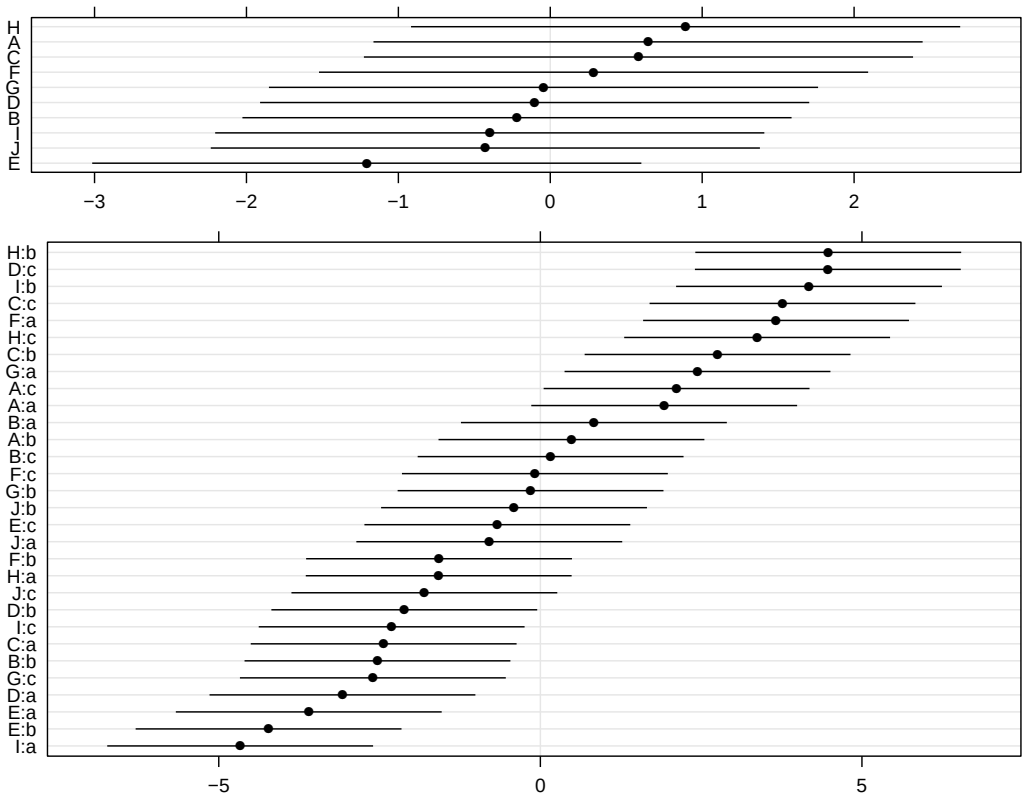


Fig. 2.10 95% prediction intervals on the random effects for model `fm2` fit to the `Penicillin` data.

clusion from Fig. 2.8 was that there may not be a significant batch-to-batch variability in addition to the sample-to-sample variability.

Plots of the prediction intervals of the random effects (Fig. **??**)) confirm this impression in that all the prediction intervals for the random effects for `batch` contain zero. Furthermore, the profile zeta plot (Fig. 2.11) shows that the even the 50% profile-based confidence interval on σ_2 extends to zero.

2.2.4 Testing $H_0 : \sigma_2 = 0$ Versus $H_a : \sigma_2 > 0$

One of the many famous quotes from Albert Einstein is "Everything should be made as simple as possible, but not simpler." In statistical modeling this *principal of parsimony* is embodied in hypothesis test comparing two models, one of which contains the other as a special case. Typically, one or more of the parameters in the more general model, which we call the *alternative hypothesis*, is constrained in some way, resulting in the restricted model, which we call the *null hypothesis*. Although we phrase the hypothesis test in terms

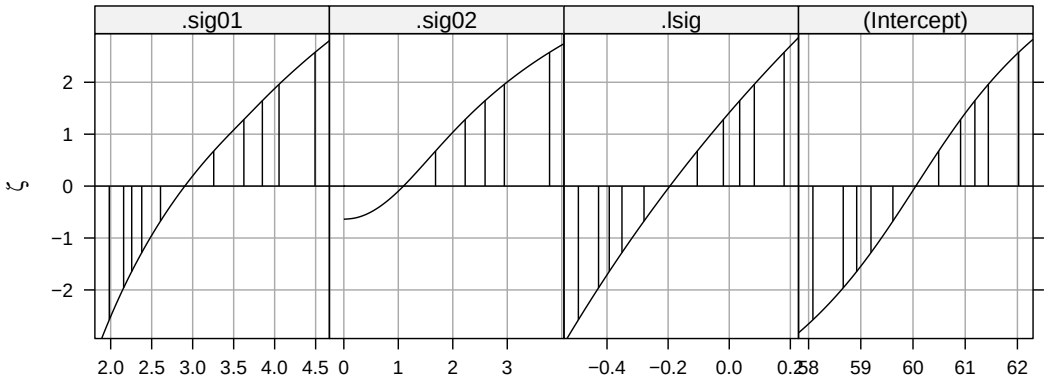


Fig. 2.11 Profile zeta plots for the parameters in model `fm3`.

of the parameter restriction, it is important to realize that we are comparing the quality of fits obtained with two different models.

Because the more general model, H_a, must provide a fit that is at least as good as the restricted model, H_0, our purpose is to determine whether the change in the quality of the fit is sufficient to justify the greater complexity of model H_0. This comparison is often reduced to a *p-value* which is the probability of seeing a difference in the model fits as large as we did, or even larger, when, in fact, H_0 is adequate. Like all probabilities, a p-value must be between 0 and 1. When the p-value for a test is small (close to zero) we prefer the more complex model, saying that we "reject H_0 in favor of H_a". On the other hand, when the p-value is not small we "fail to reject H_0", arguing that there is a non-negligible probability that the observed difference in the model fits could reasonably be the result of random chance, not the inherent superiority of the model H_a. Under these circumstances we prefer the simpler model, H_0 according to the principal of parsimony.

These are the general principles of statistical hypothesis tests. To perform a test in practice we must specify the criterion for comparing the model fits, the method for calculating the p-value from the observed value of the criterion and the standard by which we will determine if the p-value is "small" or not. The criterion is called the *test statistic*, the p-value is calculated from a *reference distribution* for the test statistic, and the standard for small p-values is called the *level* of the test.

In Sect. 1.5 we referred to likelihood ratio tests (LRTs) for which the test statistic is the difference in the deviance. That is, the LRT statistic is $d_0 - d_a$ where d_a is the deviance in the more general (H_a) model fit and d_0 is the deviance in the constrained (H_0) model. An approximate reference distribution for an LRT statistic is the χ^2_ν distribution where ν, the degrees of freedom, is determined by the number of constraints imposed on the parameters of H_a to produce H_0.

The restricted model fit

```
> (fm3a <- lmer(strength ~ 1 + 1|sample, Pastes, REML=0))

Linear mixed model fit by maximum likelihood
Formula: strength ~ 1 + 1 | sample
   Data: Pastes
   AIC   BIC logLik deviance
 254.4 260.7 -124.2    248.4

Random effects:
 Groups   Name        Variance Std.Dev.
 sample   (Intercept) 9.6328   3.1037
 Residual             0.6780   0.8234
Number of obs: 60, groups: sample, 30

Fixed effects:
            Estimate Std. Error t value
(Intercept)  60.0533     0.5765   104.2
```

is compared to model `fm3` with the `anova` function

```
> anova(fm3a, fm3)

Data: Pastes
Models:
fm3a: strength ~ 1 + 1 | sample
fm3: strength ~ 1 + (1 | sample) + (1 | batch)
     Df    AIC    BIC  logLik  Chisq Chi Df Pr(>Chisq)
fm3a  3 254.40 260.69 -124.20
fm3   4 255.99 264.37 -124.00 0.4072      1     0.5234
```

which provides a p-value of 0.5234.Because typical standards for "small" p-values are 5% or 1%, a p-value over 50% would not be considered significant at any reasonable level.

We do need to be cautious in quoting this p-value, however, because the parameter value being tested, $\sigma_2 = 0$ is on the boundary of set of possible values, $\sigma_2 \geq 0$, for this parameter. The argument for using a χ^2_1 distribution to calculate a p-value for the change in the deviance does not apply when the parameter value being tested is on the boundary. As shown in Pinheiro and Bates [2000, Sect. 2.5], the p-value from the χ^2_1 distribution will be "conservative" in the sense that it is larger than a simulation-based p-value would be. In the worst-case scenario the χ^2-based p-value will be twice as large as it should be but, even if that were true, an effective p-value of 26% would not cause us to reject H_0 in favor of H_a.

2.2.5 Assessing the Reduced Model, `fm3a`

The profile zeta plots of the parameters in model `fm3a` (Fig. **??**) are similar

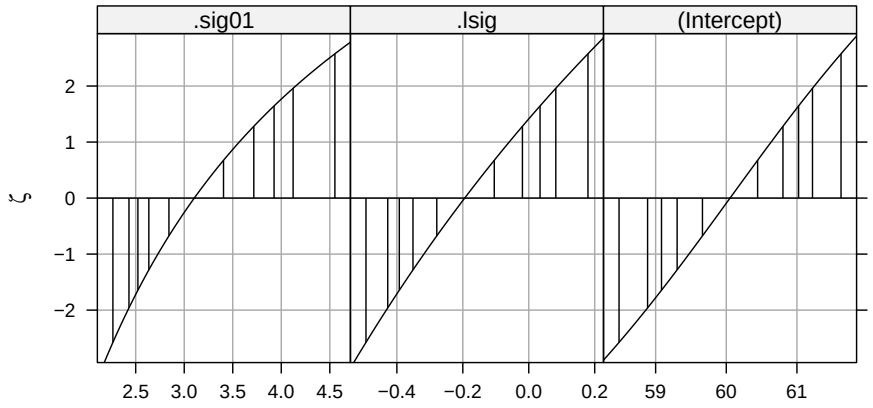


Fig. 2.12 Profile zeta plots for the parameters in model `fm3a`.

to the corresponding plots in Fig. **??**, as confirmed by the numerical values of the confidence intervals.

```
> confint(pr3)
```

```
                  2.5 %      97.5 %
.sig01       2.1579337  4.05358895
.sig02              NA  2.94658928
.lsig       -0.4276761  0.08199287
(Intercept) 58.6636504 61.44301637
```

```
> confint(pr3a)
```

```
                  2.5 %      97.5 %
.sig01       2.4306377  4.12201052
.lsig       -0.4276772  0.08199277
(Intercept) 58.8861831 61.22048353
```

The confidence intervals on $\log(\sigma)$ and β_0 are similar for the two models. The confidence interval on σ_1 is slightly wider in model `fm3a` than in `fm3`, because the variability that is attributed to `batch` in `fm3` is incorporated into the variability due to `sample` in `fm3a`.

The patterns in the profile pairs plot (Fig. 2.13) for the reduced model `fm3a` are similar to those in Fig. 1.9, the profile pairs plot for model `fm1`.

2.3 A Model With Partially Crossed Random Effects

Especially in observational studies with multiple grouping factors, the configuration of the factors is neither nested nor completely crossed. We describe such situations as having *partially crossed* grouping factors for the random effects.

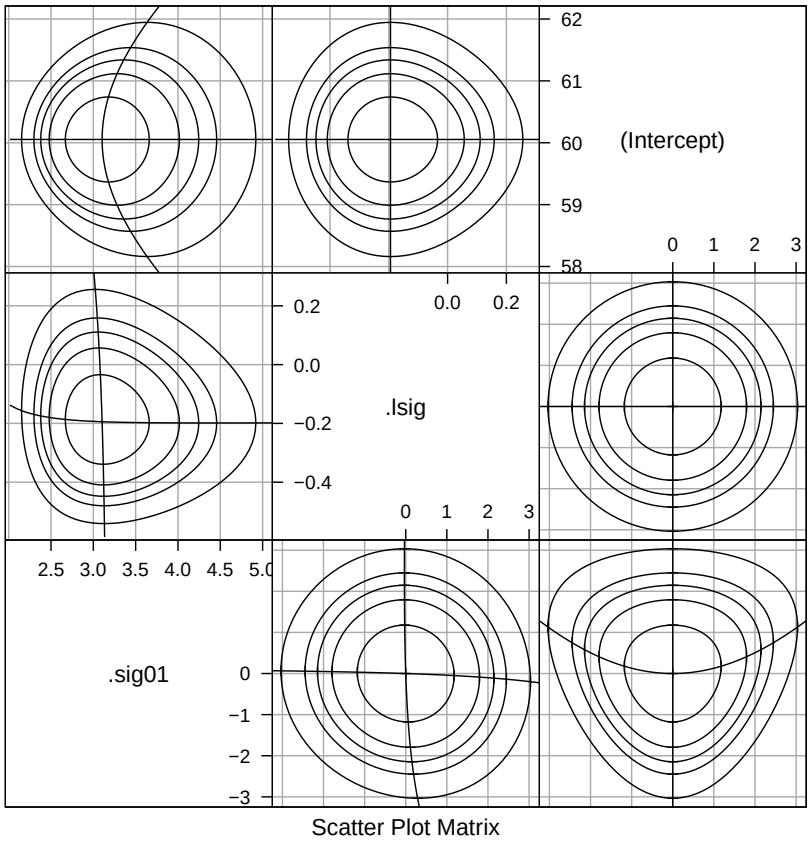


Fig. 2.13 Profile pairs plot for the parameters in model `fm3a` fit to the `Pastes` data.

Studies in education, in which test scores for students over time are also associated with teachers and schools, usually result in partially crossed grouping factors. When students with scores in different years have different teachers for the different years, the student factor is not nested within the teacher factor. To have complete crossing of the student and teacher factor is would be necessary for each student to be observed with each teacher which would be unusual. A study of thousands of students and hundreds of teachers inevitably ends up partially crossed.

In this section we consider an example with thousands of students and instructors where the response is the student's evaluation of the instructor's effectiveness. These data, like those from most large observational studies, are quite unbalanced.

2.3.1 The `InstEval` *Data*

The `InstEval` data are from a special evaluation of lecturers by students at ETH-Zürich, to determine who should receive the "best-liked professor"

award. These data have been slightly simplified and identifying labels removed so as to preserve anonymity.

The variables

```
> str(InstEval)

'data.frame':        73421 obs. of  7 variables:
 $ s      : Factor w/ 2972 levels "1","2","3","4",..: 1 1 1 1 2 2 3 3 3 3 ...
 $ d      : Factor w/ 1128 levels "1","6","7","8",..: 525 560 832 1068 62 406 3 6 19 75 ...
 $ studage: Ord.factor w/ 4 levels "2"<"4"<"6"<"8": 1 1 1 1 1 1 1 1 1 1 ...
 $ lectage: Ord.factor w/ 6 levels "1"<"2"<"3"<"4"<..: 2 1 2 2 1 1 1 1 1 1 ...
 $ service: Factor w/ 2 levels "0","1": 1 2 1 2 1 1 2 1 1 1 ...
 $ dept   : Factor w/ 14 levels "15","5","10",..: 14 5 14 12 2 2 13 3 3 3 ...
 $ y      : int  5 2 5 3 2 4 4 5 5 4 ...
```

have somewhat cryptic names. Factor `s` designates the student and `d` the instructor. The `dept` factor is the department for the course and `service` indicates whether the course was a service course taught to students from other departments.

Although the response, `y`, is on a scale of 1 to 5,

```
> xtabs(~ y, InstEval)

y
    1     2     3     4     5
10186 12951 17609 16921 15754
```

it is sufficiently spread out to warrant treating is as if it were a continuous response.

In this chapter we fit models that have random effects for student, instructor, and department or the `dept:service` combination to these data. Later we will fit models that have fixed for instructor and department to these data.

```
> (fm4 <- lmer(y ~ 1 + (1|s) + (1|d) + (1|dept:service), InstEval, REML=0))

Linear mixed model fit by maximum likelihood
Formula: y ~ 1 + (1 | s) + (1 | d) + (1 | dept:service)
   Data: InstEval
    AIC    BIC  logLik deviance
 237663 237709 -118827   237653

Random effects:
 Groups       Name        Variance Std.Dev.
 s            (Intercept) 0.105405 0.32466
 d            (Intercept) 0.262556 0.51240
 dept:service (Intercept) 0.012133 0.11015
 Residual                 1.384953 1.17684
Number of obs: 73421, groups: s, 2972; d, 1128; dept:service, 28

Fixed effects:
            Estimate Std. Error t value
(Intercept)  3.25521    0.02824   115.3
```

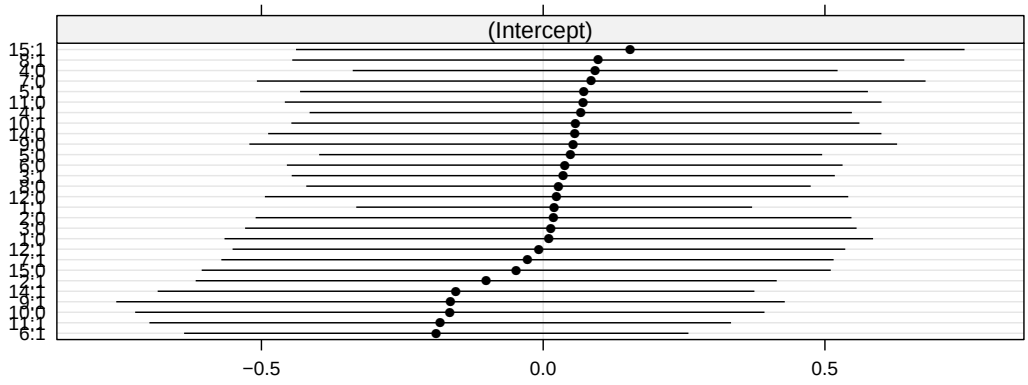


Fig. 2.14 95% prediction intervals on the random effects for the `dept:service` factor in model `fm4` fit to the `InstEval` data.

(Fitting this complex model to a moderately large data set takes less than two minutes on a modest laptop computer purchased in 2006. Although this is more time than required for earlier model fits, it is a remarkably short time for fitting a model of this size and complexity. In some ways it is remarkable that such a model can be fit on a modest laptop.)

All three estimated standard deviations of the random effects are less than $\widehat{\sigma}$, with $\widehat{\sigma_3}$ the estimated standard deviation of the random effects for the `dept:service` interaction less than one-tenth the estimated residual standard deviation.

It is not surprising that zero is within all of the prediction intervals on the random effects for this factor (Fig. 2.14). In fact, zero is close to the middle of all these prediction intervals.

References

Douglas M. Bates and Donald G. Watts. *Nonlinear Regression Analysis and Its Applications*. Wiley, Hoboken, NJ, 1988. ISBN 0-471-81643-4.

G.E.P. Box and G.C. Tiao. *Bayesian Inference in Statistical Analysis*. Addison-Wesley, Reading, MA, 1973.

Owen L. Davies and Peter L. Goldsmith, editors. *Statistical Methods in Research and Production*. Hafner, 4th edition, 1972.

Friedrich Leisch. Sweave: Dynamic generation of statistical reports using literate data analysis. In Wolfgang Härdle and Bernd Rönz, editors, *Compstat 2002 — Proceedings in Computational Statistics*, pages 575–580. Physica Verlag, Heidelberg, 2002. URL `http://www.stat.uni-muenchen.de/~leisch/Sweave`. ISBN 3-7908-1517-9.

José C. Pinheiro and Douglas M. Bates. *Mixed-effects Models in S and S-PLUS*. Springer, 2000.

J. Rasbash, W. Browne, H. Goldstein, M. Yang, and I. Plewis. *A User's Guide to MLwiN*. Multilevel Models Project, Institute of Education, University of London, London, 2000.

Stephen W. Raudenbush and Anthony S. Bryk. *Hierarchical Linear Models: Applications and Data Analysis Methods*. Sage, 2nd edition, 2002. ISBN 0-7619-1904-X.

Y. Sakamoto, M. Ishiguro, and G. Kitagawa. *Akaike Information Criterion Statistics*. Reidel, Dordrecht, Holland, 1986.

Deepayan Sarkar. *Lattice: Multivariate Data Visualization with R*. Springer, 2008.

G. Schwarz. Estimating the dimension of a model. *Annals of Statistics*, 6:461–464, 1978.