MAS474/MAS6003 Extended Linear Models

Module Information

1. Please use the discussion boards on MOLE for asking questions. This is so that everyone has the same information and receives the same level of help. If you email me with a question about the course, and it is a question that is relevant to others (e.g., *I don't understand page 2, what does x mean?, is there a typo on page 7?, the coursework question is ambigious* etc), then I will ask you to post your question to the MOLE discussion board before I answer it (note that you can post questions anonymously if you wish). I receive an email when posts appear on MOLE, and will try to answer all queries as soon as I can. Posting them on MOLE also means that other students can help to answer your questions. It is said that you don't really learn something until you've had to teach it - helping to answering other students' questions on MOLE can help cement your own understanding. Finally, a good strategy for asking questions is to be specific. Vague questions such as, '*I don't understand chapter 5*', '*I'm stuck with the coursework*' etc, can only be answered with follow-up questions from me. It is much better to say, '*I've read through chapter 5 multiple times and I'm confused. I can't see how equation 4 follows from equation 5, and I don't understand why the model proposed in section 2.11 is a sensible one?*' etc.

2. Please see the discussion boards for information about my office hours.

3. There is a project, worth 30% of the assessment for MAS474, and 15% of the assessment for MAS6003. More details and deadlines will be posted on MOLE.

4. There are two sets of non-assessed exercises. More details are available on MOLE.

5. There are further exercises ("Tasks") distributed throughout the notes. You should attempt these as we reach them during the semester. Solutions are not given (typically the tasks involve working through examples in the notes), but you may ask for help with the tasks at any time.

Contents

Chapter 1

Introduction

1.1 Statistical Models

A statistical model for data $\boldsymbol{y} = (y_1, \ldots, y_n)'$ is a description of the way that $\boldsymbol{y}$ could have arisen, usually taking account of the fact that other values than $\boldsymbol{y}$ itself could have been observed. The fundamental statistical model, that on which much of modern statistical practice rests, identifies $\boldsymbol{y}$ with the realized value of a random vector $\boldsymbol{Y}$. This immediately connects Statistics and Probability, making it natural to formulate statistical questions as questions about the properties of probability distributions.

An important special case, often appropriate when repeated observations are made under essentially identical conditions, supposes that the individual observations y_i are realizations of independent random variables Y_i, each with the same distribution. Thus a random sample is represented in terms of a set of independent, identically distributed random variables, statistical questions are formulated in terms of their common distribution and methods of inference can be constructed and evaluated accordingly. This is the basis for most of the elementary methods studied in the Foundation Module of the MSc course and in early units of the undergraduate course.

In many statistical investigations, however, there is interest in the relationship between different variables. A common situation, for example, is that in which we wish to predict one variable from known values of others. Another is when we seek understanding of how one variable depends on others. In both cases one variable is regarded as a response variable and the others as explanatory variables. This unit concentrates on such situations. (MAS6011 considers situations when variables are regarded more symmetrically.) In this case a statistical model must express a definite structure which explains the values ob-

served. Suppose that y_i denotes the ith value of the response variable and $\boldsymbol{x}_i = (x_{i1}, \ldots, x_{ip})$ values of the corresponding explanatory variables. A very general structure would represent y_i as the realized value of a random variable Y_i with a distribution depending on $\boldsymbol{x}_i$ in an arbitrarily complex way, but for most practical purposes simple structures suffice, and have the important benefit of tractability. A simple basic structure represents Y_i as the sum of a systematic (non-random) part and a random part:

$$Y_i = \text{systematic part} + \text{random part} \tag{1.1}$$

in which the systematic part depends on $\boldsymbol{x}_i$ and the random part may or may not depend on $\boldsymbol{x}_i$. We will not here attempt to account for the fact that in some applications the values of explanatory variables could themselves have been different; we will argue conditionally on the values actually observed.

1.2 Normal Linear Model

In the Normal Linear Model (or *General Linear Model*) the systematic part is the sum of a number of systematic components, and the random part has a normal distribution:

$$Y_i = \sum_{j=1}^{p} x_{ij}\beta_j + \epsilon_i, \quad i = 1, \ldots, n \tag{1.2}$$

where $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)'$ are unknown parameters and the ϵ_i are $N(0, \sigma^2)$ random variables. It follows that Y_i is normally distributed with expectation $\sum x_{ij}\beta_j$ and variance σ^2. In many applications it is reasonable to assume that the Y_i are independent.

This model has been studied in depth in MAS6003(1)/MAS363. Particularly important topics are

- Properties of estimates;
- Likelihood;
- Confidence intervals and tests;
- Categorical variables;
- Diagnostics;
- Transformations.

1.3 Topics for this module

In this module, we will study two extensions to the normal linear model:

1. mixed effects models to account for correlated errors/grouping structure in the data;
2. dealing with missing data.

We will spend 10 lectures on each topic. In the next section, we give a motivating example for mixed effect models.

1.4 Mixed effects models: an example

Three varieties of potatoes were grown to examine how their yield is influenced by adding manure and potash to the soil. The experiment was performed on a field divided into two large blocks. Each block was divided into 18 plots, each plot being given one of the different combinations of variety (A, B or C), manure (none or some) and potash (none, one unit per hectare or two units per hectare). The yields on each of the 36 plots are shown below. The two figures in each cell are for the two blocks, with the yield for block 1 given first.

		Potash level					
Variety	Manure	Zero		1 unit		2 units	
A	None	12.1	11.4	14.1	19.0	17.5	19.6
	Some	17.0	22.5	19.4	22.8	27.2	23.0
B	None	15.8	16.4	23.0	21.4	22.2	21.6
	Some	17.2	25.3	20.1	29.0	26.2	26.5
C	None	1.6	9.1	11.2	13.9	10.0	12.0
	Some	10.5	13.7	19.1	19.4	15.6	24.7

In this example interest lies in the possible relationship of Yield, the response variable, with Variety, Manure and Potash, and there is a possibility, not of direct interest in itself but of potential relevance in an attempt to represent the structure of the data accurately, that Block may have an effect.

The effect of Block may similarly be taken into account in the systematic part of the linear model, by including appropriate x_{ij} and associated parameter terms in the sum in (1.2). To do so would allow us to estimate differences between the expected yields of crops grown in blocks 1 and 2 of the field, and this

would be important if we were interested in quantifying fertility differences in this specific field, perhaps with a view to growing potatoes there again. However if the main interest is in comparison of varieties and how much manure and potash to apply, and our hope is that results will translate to other growing locations, then we might prefer to regard the effect of the block in which a plant grew as affecting its yield in a random way that might have been different in a different field, but similar for neighbouring plants. This suggests reflecting the effect of block in our model through the random part rather than through the systematic part.

A simple way to incorporate the ideas above into a linear model is to suppose that the block in which a plant grows has the potential to affect its yield, but does so by an amount that might differ between blocks and between repetitions of the experiment. Thus we might suppose that

$$Y_i = \sum_{j=1}^{p} x_{ij}\beta_j + b_{block(i)} + \epsilon_i, \quad i = 1, \ldots, n \tag{1.3}$$

where $block(i)$ denotes the block in which the ith plant grew. In this case there are two terms, b_1 and b_2, representing the effect of conditions in the two blocks. We suppose that b_1 and b_2 are independent $N(0, \sigma_b^2)$ random variables for some variance σ_b^2 representing the typical variability in fertility etc between blocks, and ϵ_i are independent $N(0, \sigma^2)$ random variables resulting from within-block sources of variability, independent of block location. The terms in the sum in (1.3) represent the systematic effects of variety, manure and fertilizer as before.

The difference between models (1.2) and (1.3) lies in the random part. In (1.2) it was given by the ϵ_i variables alone, but in (1.3) it is given by $b_{block(i)} + \epsilon_i$ and therefore has richer structure. Note for example that, from (1.3), $\text{var}(Y_i) = \sigma_b^2 + \sigma^2$ and

$$\text{Cov}(Y_i, Y_j) = \sigma_b^2, \quad \text{if } i \neq j \text{ and } i, j \text{ belong to the same block.}$$

Model (1.3) is an example of a mixed effects model.

Chapter 2

Mixed Effects Models: Basic Theory and Model Fitting in R

2.1 Introduction

This course is concerned with extensions to the standard normal-theory linear model. In the standard model all observations are taken to be **independent**; the extension in the first half of the module is to the commonly occurring situation in which observations arise in **groups** and within each group the observations are **correlated**.

Common examples include:

- **Blocked** experiments in which the design recognizes that certain units are similar (eg twins or adjacent plots in a field) and so responses from them are likely to be correlated. Sometimes there are multiple levels of grouping and an associated hierarchy of correlations, eg classes within schools within type of school within regions.
- **Repeated measures** experiments in which we take multiple responses from each subject and these are expected to be linked; for example, by growth curves.

The approach we shall take to modelling such features is to recognize that the variables that define the grouping structure of the data should be treated differently from the other explanatory variables in our model (the so-called **fixed effects**). In addition to whatever model we employ to explain the influence of the fixed effects on the response, we should include additional **random**

effects terms to account for the grouping. Thus our model will have the general form

$$\text{response} = \underset{\substack{\text{for experimental} \\ \text{factors}}}{\text{fixed effects}} + \underset{\substack{\text{for grouping} \\ \text{factors}}}{\text{random effects}} + \text{within group error}$$

In any modelling exercise, we are attempting to attribute as much as possible of the variability observed in the study to specific causes. If we use ordinary fixed effects terms in our model to account for groups, we are essentially removing the effects of the different data groups from the noise in the study. If the groups are unique to this study and there would be a different set of groups in any new version, then we are pretending accuracy we do not have. Another way of looking at the issue is that if we ignore the correlation between items in the data set and treat them all as independent, then we are deluding ourselves that we have more information than is truly available. Thus we recognize that we must use effects of a different nature to account for the groups.

Essentially we regard the particular groups used in our experiment as a **random selection** from a **population of possible groups**. Each influences the response in an additive way, but we are not interested in these particular effects. Instead our interest lies in the variability of the population from which they are drawn, since this gives an indication of the range of grouping conditions that might be encountered and how much the mean response might vary in response to these different conditions.

Random effects represent sources of variation in a model additional to that in the experimental units. Fitting a model ignoring random effects when they should be present will result in mis-estimation of the variance of the experimental units. This in turn is likely to invalidate or at best cast doubt on inferences. Hence it is vital to account for grouping of data not only at the design stage of an experiment, but also in its analysis.

The models we will be using are often termed **hierarchical models** because of the hierarchy of sources of error—at the finest level we have the ordinary errors in individual experimental units (termed within-group or residual errors), but superimposed on top of those we have the extra errors associated with random effects representing successively coarser levels of grouping. Other names include:

- **multilevel models** – a synonym.

- **random effects models** – because random effects are a common device for accounting for the grouping structure.
- **mixed effects models** (as in R and these notes) – since they include both fixed and random effects (the mean is usually taken as fixed even in an otherwise purely random model).
- **type II models** (in the older experimental design literature) – to distinguish them from Type I, i.e. purely fixed effects, models.
- **variance components models** – recognizing that the grouping structure introduces multiple levels of error (represented by variances) into our data and model.
- **repeated measures models** – reflecting a common applications area.
- **longitudinal data models** – a synonym for the above.

It is helpful to know that these terms refer to essentially the same class of models. However the approach, emphasis, terminology and notation are not standard across the different literatures.

Hierarchical/mixed effects modelling has gained in popularity recently with the advent of powerful software for model fitting. The approach taken here presents these models as extensions of the basic lm. Familiarity with the basic lm model structure, fitting procedure, estimation theory, distributional properties of estimators and diagnostics is assumed.

2.2 Three approaches to fitting mixed effects models

Three different ways of fitting mixed effects models are

1. the "classical" approach, based on sums of squares[1];
2. likelihood-based methods;
3. Bayesian methods.

In this module, we will mainly study likelihood methods, which we will be using for all practical work. Likelihood methods are

[1] See for example Cochran, W. G. and Snedecor, G. W. (1989). Statistical Methods (8th edn.), Iowa State University Press.

more flexible than the classical approach. However, to help understand these models, and the implications of switching from fixed to random effects, I think it's easier to first look at the classical approach. Bayesian modelling is another option, but requires a good understanding of Bayesian statistics first!

2.3 A motivating example and the classical approach

Fixed effects

Suppose we are comparing two drugs, and we have J patients. Each patient receives each drug (on separate occasions), and K replicate measurements are taken. Measurement errors are assumed to be independent. We consider the model

$$Y_{ijk} = \mu + \alpha_i + \beta_j + \gamma_{ij} + \varepsilon_{ijk}, \tag{2.1}$$

for $i = 1, 2$ representing drug, $j = 1, \ldots, J$ representing patient, and $k = 1, \ldots, K$ representing replicate, with $\varepsilon_{ijk} \sim N(0, \sigma^2)$. We use 'sum-to-zero' constraints

$$\sum_{i=1}^{2} \alpha_i = \sum_{j=1}^{J} \beta_j = \sum_{i=1}^{2} \gamma_{ij} = \sum_{j=1}^{J} \gamma_{ij} = 0.$$

Note that this leaves $1 + (2-1) + (J-1) + (2-1)(J-1) = 2J$ parameters. The interaction terms γ_{ij} are necessary if the expected change from switching from drug 1 to drug 2 is different for each of the J patients (a likely scenario in reality). Without the interaction, the model predicts the same change in every patient when switching from drug 1 to drug 2.

The least-squares estimator for the treatment effect parameter α_1 is

$$\hat{\alpha}_1 = \frac{\overline{Y}_{1\bullet\bullet} - \overline{Y}_{2\bullet\bullet}}{2}, \tag{2.2}$$

where

$$\overline{Y}_{i\bullet\bullet} = \frac{\sum_{j=1}^{J} \sum_{k=1}^{K} Y_{ijk}}{JK}. \tag{2.3}$$

Then

$$\hat{\alpha}_1 \sim N\left(\alpha_1, \frac{\sigma^2}{2JK}\right). \tag{2.4}$$

Now suppose we have $J = 2$ and consider taking more replicate observations per patient, by increasing K. As K increases, $Var(\hat{\alpha})$ decreases, so we should expect to learn the true treatment effect (the width of a 95% confidence interval for α_1 will

be very small if K is large). But can this be right? Can we really learn how good the treatment from observations on only two patients?

Firstly, we need think carefully about how to interpret α_1 (I was deliberately vague earlier!). From (2.2) and (2.4) we have

$$E(\overline{Y}_{1\bullet k} - \overline{Y}_{2\bullet k'}) = 2\alpha_1, \tag{2.5}$$

so that $2\alpha_1$ is the expected change in response from switching from drug 1 to drug 2, averaged over *the patients within the study*. This is not the same thing as the expected change in response from switching from drug 1 to drug 2, averaged over *the whole patient population*, presumably the real quantity of interest. If we denote this population treatment effect by α_1^P, then, assuming the patients are chosen at random, we might argue that $\hat{\alpha}_1$ is an unbiased estimator of α_1^P, but we can't learn α_1^P with certainty if J is small, no matter how large K is.

Mixed effects models

If we want to consider uncertainty about the population effect α_1^P, *we have to model variability in the population*; we need to treat patient effects as random. To simplify the algebra, we'll start with an additive model:

$$Y_{ijk} = \mu + \alpha_i^P + b_j + \varepsilon_{ijk},$$

with $b_j \sim N(0, \sigma_b^2)$, and everything else as before. Note that

1. we have modelled explicitly *two sources of random variation*: variation between measurements on the same patient, and variation between patients;
2. we do *not* state that $\sum_{j=1}^{J} b_j = 0$;
3. α_i^P really does represent population treatment effect, as b_j represents the effect of a patient randomly drawn from the population;
4. if this model were true, the treatment effect would be the same for everyone, so we would actually only need 1 patient (and large K) to learn α_i^P.

Classical approach to model fitting

We have introduced a second variance parameter, which we need to estimate. The **key idea in the classical approach** is to

find a function of the data that has expectation σ_b^2, and we can find such a function by considering a decomposition of the sum of squares.

Consider the more general case $i = 1, \ldots, I$ with the constraint $\sum_{i=1}^{I} \alpha_i^P = 0$. We can see that

$$\overline{Y}_{\bullet j \bullet} = \frac{\sum_{i=1}^{I}\sum_{k=1}^{K}(\mu + \alpha_i^P + b_j + \varepsilon_{ijk})}{IK} = \mu + b_j + \frac{\sum_{i=1}^{I}\sum_{k=1}^{K}\varepsilon_{ijk}}{IK}, \tag{2.6}$$

since we have the constraint $\sum_{i=1}^{I} \alpha_i^P = 0$. Therefore $\overline{Y}_{\bullet 1\bullet}, \ldots, \overline{Y}_{\bullet J\bullet}$ is a sample of independent normally distributed variables with mean μ and variance $\tau^2 = \sigma_b^2 + \sigma^2/IK$. We can thus recognise

$$\frac{1}{J-1}\sum_{j=1}^{J}(\overline{Y}_{\bullet j\bullet} - \overline{Y}_{\bullet\bullet\bullet})^2$$

as the usual unbiased estimator of a variance τ^2. Therefore,

$$E\left(\frac{1}{J-1}\sum_{j=1}^{J}(\overline{Y}_{\bullet j\bullet} - \overline{Y}_{\bullet\bullet\bullet})^2\right) = \tau^2 \tag{2.7}$$

$$= \sigma_b^2 + \frac{\sigma^2}{IK}. \tag{2.8}$$

Finally, if we estimate σ^2 by the usual estimator $\hat{\sigma}^2 = RSS/(n-p)$, we have an unbiased estimator of σ_b^2:

$$\hat{\sigma}_b^2 = \frac{1}{J-1}\sum_{j=1}^{b}(\overline{Y}_{\bullet j\bullet} - \overline{Y}_{\bullet\bullet\bullet})^2 - \frac{\hat{\sigma}^2}{IK}. \tag{2.9}$$

Note that this estimator is not guaranteed to be positive! This is one reason why we will be using likelihood methods later on.

Mixed effects model with interactions

Now we return to the full model with interactions. Again, we think of the patients in the study as a random sample of patients, so that the interaction parameters are also random. We write

$$Y_{ijk} = \mu + \alpha_i^P + b_j + b_{ij} + \varepsilon_{ijk},$$

with $b_j \sim N(0, \sigma_{b_1}^2)$, and $b_{ij} \sim N(0, \sigma_{b_2}^2)$, with no constraints placed on either the b_j or b_{ij} parameters. If we consider the estimator of α_1^P:

$$\hat{\alpha}_1^P = \frac{\overline{Y}_{1\bullet\bullet} - \overline{Y}_{2\bullet\bullet}}{2}, \tag{2.10}$$

the b_j terms cancel out, but the interaction terms b_{ij} do not, and we can show that

$$\hat{\alpha}_1^P \sim N\left(\alpha_1^P, \frac{\sigma^2}{2JK} + \frac{\sigma_{b_2}^2}{2J}\right). \tag{2.11}$$

We can now see that to make the variance of the estimator small, assuming $\sigma_{b_2}^2$ is non negligible, we need the number of patients J to be large; simply taking lots of replicates (K large) on a small number of patients will not work. Of course, this is common sense, but the important thing is that it is the random effects model that gives the correct framework for dealing with the variation in the data.

Task 1. *Download and work through the example* `MAS474-classical.R`. *This uses simulated data, so you can compare estimates with true parameter values.*

Having (briefly) studied the classical approach, we will now move on to likelihood methods. I think these are harder to understand in depth, but not too difficult to use in practice. Likelihood methods are more flexible (for example in dealing with unbalanced designs), and avoid the problem of negative variance estimators.

2.4 Mixed effects models in R and textbooks

The current state of play is a little confusing! Two choices are

- `nlme`, described in Pinhiero & Bates (2000)[2]
- `lme4`, in part described in Bates (2010)[3]. However, the current version of `lme4` on the CRAN site (which is what you will get if you install `lme4` in the usual way using the R gui) differs from the version described in Bates (2010).

(Note that Professor Bates is an author of both R packages). `lme4` is the more up to date package, and the one that we will use in this module, but Pinhiero & Bates (2000) is still a good book for learning more about the underlying theory. A shorter (and

[2]Pinheiro, J. C. and Bates, D. M. (2000). Mixed-Effects Models in S and S-PLUS, Springer-Verlag New York.

[3]Bates, D. M. (2010). lme4: Mixed-effects modelling with R. Unpublished, available at `http://lme4.r-forge.r-project.org/book/`, accessed 16/1/15.

recommended) treatment (based on `lme4`) is given in Faraway (2006)[4]

2.5 Model Fitting in R: A First Example

We will first work through a simple example to explain the basic concepts of linear mixed models, how to fit them in R and to interpret the output.

2.5.1 A Simple Randomized Block Design

A simple randomized block design is one in which we have a single **experimental** factor (for which we will use fixed effects) and a **blocking** factor (for which we will use random effects).

Example: An Ergometrics Experiment. To illustrate the approach we use the data set `ergoStool` in `MAS474.RData`. The R script file `MAS474-ergoStool.R` contains the R analysis described here (you may wish to run through it whilst reading this section). These data are from an ergonomics experiment, in which 4 different stools are being compared for ease of use. The data consist of the effort required by each of 9 different subjects to rise from each stool. The subjects represent a sample of people from the general population of potential users of the stools. The objective of the experiment is to discover whether there are systematic differences in the effort needed to rise from the different stools.

First load the `lme4` library. The data set contains three variables: `effort` giving the observed values of the response variable; `Type` specifying the type of stool and taking values `T1`, `T2`, `T3` and `T4`; and `Subject` identifying the individual subjects testing the stools and taking values 1 – 9. Although `Subject` takes numerical values, it has been set up in the data set as a `factor` object. `Type` is automatically taken to be a `factor` since it has non-numerical values.

2.5.2 Inspecting the data

Some useful R commands for examining the data are

1. `str(ergoStool)`

[4]Faraway, J. J. (2006). Extending the Linear Model with R: Generalized Linear, Mixed Effects and Nonparametric Regression Models, Chapman & Hall/CRC: Boca Raton.

Display the structure of the data. Use this to check that the factor variables really are defined as factors.

2. `head(ergoStool)`
   Display the first few rows of the dataframe.

3. `tail(ergoStool)`
   Display the last few rows of the dataframe

4. `xtabs(~ Type + Subject, ergoStool)`
   Count how many observations there are for each combination of factors.

5. `matrix(by(effort,list(Type,Subject),mean),4,9)`
   Calculate the mean response for each combination of factors, and arrange into a matrix.

There are many different ways of plotting the data! Some of these commands will need `attach(ergoStool)` first. You may also need to install the `lattice` library in R first.

1. `xyplot(effort~Type|Subject,data=ergoStool)`
   Plot effort against Type separately for each subject, but within the same graphics window.

2. `plot(Type,effort)`
   A box plot of effort against type.

3. `plot(Subject,effort)`
   A box plot of effort against subject.

4. `plot(reorder(Subject,effort),effort)` A box plot of effort against subject, ordered by increasing mean effort.

5. `plot.design(ergoStool)`
   A plot of mean effort for each Subject and Type.

(I have been lazy here, but make sure to include proper axes labels where needed. See the R script file `plots.R` for examples of formatting plots.) A Trellis dot-plot of the whole data set can be produced, but the command is a little too long to list here. I have included it in the R script file.

Some plots can be seen in Figure 2.1. The plots reveal differences between the effort for different stool types and differences of comparable size between the effort of different subjects (though, of course, it is not yet possible to say whether any of these differences are larger than could be explained by chance).

We will wish to use a model which incorporates both types of variability – variability between subjects and variability within

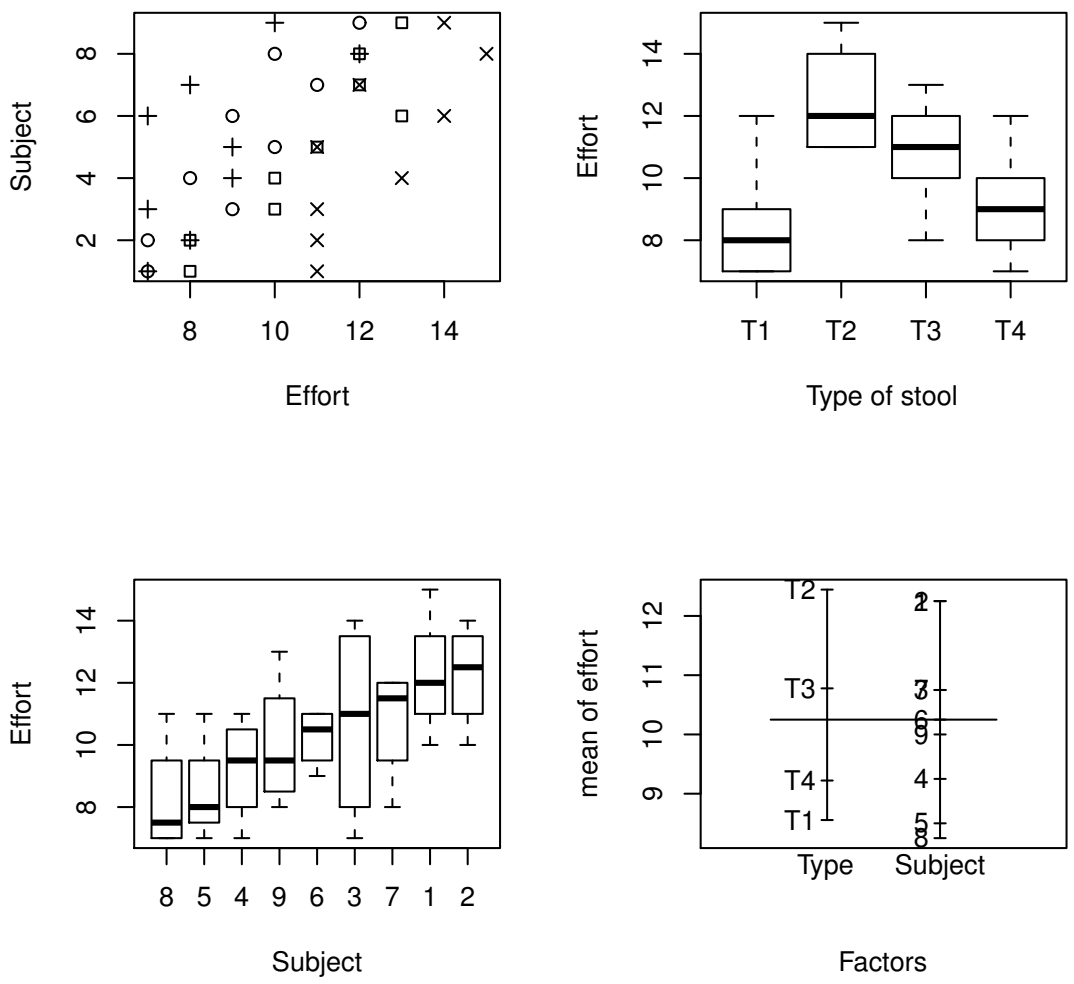

Figure 2.1: Various plots for the Ergo data

subjects – explicitly (rather than one which simply pools the subject variability with general experimental variation), as it will clearly be helpful to assign as much variability as possible to specific causes. However, we wish to treat the two types of variability differently to reflect their different status – experimental variation of direct interest, between-subject variation only of peripheral interest (perhaps for indicating the range of subject variability that a manufactured stool would face in use). We say that `Subject` is a *blocking* or *grouping* factor because it represents a known source of variability in the experiment, not of direct interest in itself but necessary to recognize to model the data accurately.

The final comment above gives the reason for our random effects approach to thinking about the subject variability – we regard the particular set of subjects used in our experiment as a **random selection** from some **population of possible subjects**. Each subject influences the mean response in some additive way, but we are not interested in the effects of these particular subjects. Instead we would like to know the overall variability in the population of possible subject effects.

This suggests the model

$$Y_{ij} = \beta_j + b_i + \epsilon_{ij}, \quad i = 1, \ldots, 9, \quad j = 1, \ldots, 4, \tag{2.12}$$

where Y_{ij} represents the effort required by the ith subject to rise from the jth type of stool, β_j is the fixed effect of the jth type of stool, b_i is the random effect of the ith subject and ϵ_{ij} is the random error for the experimental units (subject-type combination). To complete the model we need to specify the distributions of the random variables b_i and ϵ_{ij}. A natural starting point is to take these both as independent Normals with zero mean and constant variances:

$$b_i \sim N(0, \sigma_b^2), \quad \epsilon_{ij} \sim N(0, \sigma^2). \tag{2.13}$$

These assumptions can be checked (chiefly graphically) and modified if necessary.

The terminology used in describing our model now becomes clear. We have two sources of variation – the b_i and the ϵ_{ij}. The b_i represent differences of subjects from the overall response, and the ϵ_{ij} represent the natural variability associated with making the measurement; given the value of b_i – that is, given an individual subject – we would still expect to see variability from one measurement to another, and it is this variability that the ϵ_{ij} describe. The b_i are *effects* because they represent a deviation in the mean for subject i from the overall level. They are *random* effects because they are selected at random from a population of

possible subjects' effects. The model is *hierarchical* in the sense that it describes a hierarchy of effects: it could be specified in terms of a model for the response *conditional* on the subject, together with a model for choice of the subject effect:

$$\begin{aligned} Y_{ij} \,|\, b_i &\sim N(\beta_j + b_i, \sigma^2) \\ b_i &\sim N(0, \sigma_b^2). \end{aligned}$$

Because the observations on the same subject share the same random effect b_i, they are correlated. The covariance between two observations on the same subject is σ_b^2 and the corresponding correlation $\sigma_b^2/(\sigma_b^2 + \sigma^2)$. We have chosen to present our model selection through a consideration of the grouping structure of the data; however, an alternative approach is to model the correlations between observations more directly. Different sections of the literature approach the issue from these different directions, though, naturally, equivalent models can be reached.

Task 2. *Working directly from the definitions and model, verify the statements above about the covariance and correlation of observations made on the same subject.*

The parameters of the model are the $\beta_1, \ldots, \beta_4$, σ_b^2 and σ^2. Note that we have retained the convention of representing (only) parameters by Greek letters and that the number of parameters in the model will always be 6, irrespective of the number of subjects in the experiment. Although the random effects b_i behave rather like parameters, formally they are just another level of random variation in the data and so we do not estimate them as such. We do, however, form predictions of the values of these random variables, given the data we have observed.

Note that the fixed effects in the model are to be interpreted in the usual way. We have specified the model here in the *cell means* formulation (in which β_j represents what would be the mean effort required to rise from the jth type of stool if the whole population were sampled), as the β_j have a simple interpretation; however, they are not convenient to use when assessing differences between stool types. For these a **contrast**-based formulation representing a reference level and deviations from it would be preferable. Using the **treatment contrasts** we would fix the reference level as the mean value under `Type T1` and use the other three parameters to represent the differences of `Types T2, T3 and T4` from this in our model fitting.

2.5.3 Fitting the Model in R

To fit the model (2.12) in R we use the `lmer` function, giving the resulting object a name so that it will be available for further analysis later. A call to `lmer` is similar to one to `lm` to fit an ordinary linear model but has additional arguments to specify random effects. For the present model type

```
(fm1<-lmer(effort~Type - 1 + (1|Subject),data=ergoStool))
```

The basic form of the command is

`lme`(*specification of systematic part + (specification of random part), data*)

The specification of the systematic part of the model is in the same form as for `lm` (called a *linear model formula* in R):

`response variable ~ fixed explanatory variables`

Including the `-1` term gives the cell means formulation; otherwise treatment contrasts are used. The `data` argument merely specifies the data set to be used. It may be omitted if the data have been `attached` earlier.

The random part of the model is specified within the inner brackets, describing the random effects and the grouping structure. The format used above

`1 | Subject`

indicates that the random effects are simply added to the mean (denoted by `1`) and that `Subject` is the only grouping variable.

The choice of the name `fm1` was arbitrary; any other name could be used. The effect of the command `fm1 <- lmer( ...)` is to fit the mixed model and to store the results in the object `fm1`. Putting the whole command inside brackets tells R both to assign and display the results.

Task 3. *Work through the examination of the* `ergoStool` *data set and the process of fitting a simple random effects model to it.*

2.5.4 Interpreting the Output

An overview of the fitted model is obtained by typing the model name:

```
> summary(fm1)
Linear mixed model fit by REML ['lmerMod']
Formula: effort ~ Type - 1 + (1 | Subject)
   Data: ergoStool

REML criterion at convergence: 121.1
```

```
Scaled residuals:
     Min       1Q   Median       3Q      Max
-1.80200 -0.64317  0.05783  0.70100  1.63142

Random effects:
 Groups   Name        Variance Std.Dev.
 Subject  (Intercept) 1.775    1.332
 Residual             1.211    1.100
Number of obs: 36, groups:  Subject, 9

Fixed effects:
       Estimate Std. Error t value
TypeT1    8.556      0.576   14.85
TypeT2   12.444      0.576   21.60
TypeT3   10.778      0.576   18.71
TypeT4    9.222      0.576   16.01

Correlation of Fixed Effects:
       TypeT1 TypeT2 TypeT3
TypeT2 0.595
TypeT3 0.595  0.595
TypeT4 0.595  0.595  0.595
```

Firstly note that the output confirms that a linear mixed effects model has been fitted, specifies the estimation method (currently the default, REML, a variant on ordinary maximum likelihood to be discussed in Section 2.7) and the data set used. This confirmatory information can be useful for keeping track of the results of multiple fitting attempts. The `REML criterion at convergence` gives -2 times the value of the maximised REML likelihood.

We will discuss residuals in more detail later (as there are more than one type for mixed effects models), but for now, `Scaled residuals` gives summary statistics for $(y_{ij} - \hat{b}_i - \hat{\beta}_j)/\hat{\sigma}$.

The next section of the output covers the random effects elements of the model. The parameter estimates for the variance components are tabulated. The square root of each estimate is also displayed under the heading `Std. Dev.` (these are not standard errors of the estimates).

Below this is the panel of fixed effects estimates. The layout of this is much the same as for any other type of regression model. The corner point parametrization has been used, so we see an estimate labelled `(Intercept)` corresponding to the first level of `Type`, and then estimates of the expected deviations from this value for each of the other three stool types. Unlike the random effects, standard errors are provided for these estimates, together

with associated df, and t-values. p-values are *not* reported for reasons we will discuss later.

Finally, correlations between fixed effects estimators are reported.

Task 4. *The fixed effect estimators in this case are simply* $\hat{\beta}_j = \bar{Y}_{\bullet j}$. *Show that* $Cor(\hat{\beta}_j, \hat{\beta}_{j'}) = 0.595$, *for* $j \neq j'$.

Two points to note here are:

- The particular contrast setting/parameter constraints chosen will affect the individual terms and their standard errors; with sum-to-zero contrasts, for example, different values will be obtained. On the other hand an overall assessment of the effect of the `Type` factor, is not affected by the parametrization.
- One of the chief motivations for using random effects models is to ensure that the estimates of the precision of the fixed effects are appropriate. When comparing models, one often finds that the estimates of the fixed effects are (nearly) identical, but their standard errors differ substantially, so one needs to be particularly aware of the Std.Error column.

2.6 Matrix notation for mixed effects models

We now give the general form of a mixed effects model in matrix notation, which we will use in the next two sections. For illustration, we consider the following example. The dataframe `raildata` contains data from an experiment involving measuring the time for an ultrasonic wave to travel the length of the rail. These travel times were measured three times on each of six rails. The six rails were selected at random and we are not interested in properties of the particular rails used in the experiment, but rather in the population from which they were drawn. We wish to obtain an estimate of the travel time for a typical rail and an indication of the variance of the entire population. Thus we might fit the model

$$\begin{aligned} y_{ij} &= \beta + b_i + \epsilon_{ij}, \quad i = 1, ..., 6, \quad j = 1, ..., 3, \\ & \qquad b_i \sim N(0, \sigma_b^2), \quad \epsilon_{ij} \sim N(0, \sigma^2), \end{aligned} \tag{2.14}$$

where y_{ij} is the travel time recorded by the jth test on the ith rail, β is the population mean travel time, b_i is the random effect of the ith rail and ϵ_{ij} is the within-rail random error. For the

unobserved travel time Y_{ij}, we have $Y_{ij} \sim N(\beta, \sigma^2 + \sigma_b^2)$, and $Cov(Y_{ij}, Y_{i'j'}) = \sigma_b^2$, if $i = i'$, and 0 otherwise, with $j \neq j'$.

The general form of any mixed effects model in matrix notation is

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{Z}\boldsymbol{b} + \boldsymbol{\epsilon},$$

where $\boldsymbol{Y}$ is the vector of observations, $\boldsymbol{\beta}$ is a vector of fixed effects, $\boldsymbol{b}$ is the vector of random effects, $\boldsymbol{X}$ and $\boldsymbol{Z}$ are the corresponding design matrices, and $\boldsymbol{\epsilon}$ is a vector of independent errors, each normally distributed with mean 0 and variance σ^2. In the rail data example, we have X an 18×1 column of 1s and $\boldsymbol{\beta} = (\beta)$. If we write $\boldsymbol{Y}$ as $(Y_{11}, Y_{12}, Y_{13}, \ldots, Y_{61}, Y_{62}, Y_{63})^T$, then we have $\boldsymbol{b} = (b_1, b_2, \ldots, b_6)^T$ and

$$\boldsymbol{Z} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

In general, we have

$$\boldsymbol{Y} \sim N(\boldsymbol{X}\boldsymbol{\beta}, V), \tag{2.15}$$

where $V = \boldsymbol{Z}Var(\boldsymbol{b})\boldsymbol{Z}^T + \sigma^2 I_n$, where I_n is the $n \times n$ identity matrix, with n the total number of observations. In the rail data example, we have $Var(\boldsymbol{b}) = \sigma_b^2 I_6$ and V an 18×18 block diagonal matrix, made up of blocks of 3×3 matrices

$$\begin{pmatrix} \sigma^2 + \sigma_b^2 & \sigma_b^2 & \sigma_b^2 \\ \sigma_b^2 & \sigma^2 + \sigma_b^2 & \sigma_b^2 \\ \sigma_b^2 & \sigma_b^2 & \sigma^2 + \sigma_b^2 \end{pmatrix}$$

2.7 Parameter estimation using REML

For an ordinary linear model, we have closed-form expressions for the parameter estimates. In the classical approach for mixed effects models, one can also derive closed-form expressions for the parameter estimates, as we did in Section 2.3. However, these expressions can give negative estimates for variances, and require balanced designs (equal numbers of observations for each combination of factors). `lme4` uses (restricted) maximum likelihood to estimate the parameters, where the maximisation is done using a numerical optimisation routine.

2.7.1 REML: restricted maximum likelihood

Standard maximum likelihood estimates of variance components are negatively biased. For a simple example, consider i.i.d normal random variables $Y_1, \ldots, Y_n$ with unknown mean μ and variance σ^2. The unbiased estimator of σ^2 is $\hat{\sigma}^2 = \sum_{i=1}^n (Y_i - \bar{Y})^2/(n-1)$, but the maximum likelihood estimator of σ^2 is $\sum_{i=1}^n (Y_i - \bar{Y})^2/n$, which has expectation $\sigma^2 \frac{(n-1)}{n}$.

The REML criterion can be defined by integrating out the 'fixed effect', μ in this case, in the likelihood:

$$L_R(\sigma^2|\boldsymbol{y}) = \int_{-\infty}^{\infty} L(\sigma^2, \mu|\boldsymbol{y}) d\mu. \tag{2.16}$$

In this example, we can evaluate the integral analytically, by making the integrand 'look like' a normal probability density function in μ, with mean $\bar{y}$ and variance σ^2/n. We then use the fact that any density function must integrate to 1. To do this manipulation, we use the result

$$\sum_{i=1}^n (y_i - \mu)^2 = \sum_{i=1}^n (y_i - \bar{y})^2 + n(\mu - \bar{y})^2. \tag{2.17}$$

We now have

$$\begin{aligned}
\int_{-\infty}^{\infty} L(\sigma^2, \mu|\boldsymbol{y}) d\mu &= \int_{-\infty}^{\infty} \frac{1}{(\sqrt{2\pi\sigma^2})^n} \exp\left\{-\frac{1}{2\sigma^2}\sum_{i=1}^n (y_i - \mu)^2\right\} d\mu && (2.18)\\
&= \frac{1}{(\sqrt{2\pi\sigma^2})^{n-1}} \exp\left\{-\frac{1}{2\sigma^2}\sum_{i=1}^n (y_i - \bar{y})^2\right\} && (2.19)\\
&\quad \times \frac{1}{\sqrt{n}} \int_{-\infty}^{\infty} \frac{\sqrt{n}}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{n}{2\sigma^2}(\mu - \bar{y})^2\right\} d\mu && (2.20)\\
&= \frac{1}{\sqrt{n}(\sqrt{2\pi\sigma^2})^{n-1}} \exp\left\{-\frac{1}{2\sigma^2}\sum_{i=1}^n (y_i - \bar{y})^2\right\} && (2.21)
\end{aligned}$$

If we now maximise $L_R(\sigma^2|\boldsymbol{y})$ with respect to σ^2, we obtain the usual unbiased estimator of σ^2. An interpretation of the REML procedure is that we have focused attention on the residuals after least squares fitting of the 'fixed effects' (the least squares estimate of μ being $\bar{y}$, so that $y_i - \bar{y}$ is the i-th residual.

2.7.2 REML for a mixed effect model

The REML approach is estimate the variance parameters using the REML criterion, and then estimate the fixed effects by (generalised) least squares, conditional on the variance estimates.

The REML criterion is

$$L_R(\boldsymbol{\theta}, \sigma^2|\boldsymbol{y}) = \int L(\boldsymbol{\theta}, \sigma^2, \boldsymbol{\beta}|\boldsymbol{y}) d\boldsymbol{\beta}, \tag{2.22}$$

where $\boldsymbol{y}$ is the observed value of $\boldsymbol{Y}$, and θ is the variance parameters for the random effect terms. The likelihood function is obtained from the distribution of $\boldsymbol{Y}$, given in (2.15):

$$L(\boldsymbol{\theta}, \sigma^2, \boldsymbol{\beta}|\boldsymbol{y}) = \frac{|V|^{-\frac{1}{2}}}{(2\pi)^{\frac{n}{2}}} \exp\left\{-\frac{1}{2}(\boldsymbol{y} - X\boldsymbol{\beta})^T V^{-1}(\boldsymbol{y} - X\boldsymbol{\beta})\right\}, \tag{2.23}$$

where n is the number of observations. Recall that V is constructed from the variance parameters in $\boldsymbol{\theta}$. (For the rail data example, we have $\boldsymbol{\theta} = \{\sigma_b^2\}$). It is straightforward to verify that

$$(\boldsymbol{y}-X\boldsymbol{\beta})^T V^{-1}(\boldsymbol{y}-X\boldsymbol{\beta}) = (\boldsymbol{y}-X\hat{\boldsymbol{\beta}})^T V^{-1}(\boldsymbol{y}-X\hat{\boldsymbol{\beta}})+(\hat{\boldsymbol{\beta}}-\boldsymbol{\beta})^T X^T V^{-1} X(\hat{\boldsymbol{\beta}}-\boldsymbol{\beta}), \tag{2.24}$$

by noting that

$$\begin{aligned}(X\hat{\boldsymbol{\beta}} - X\boldsymbol{\beta})^\top V^{-1}(\boldsymbol{y} - X\hat{\boldsymbol{\beta}}) &= (\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})^\top X^\top V^{-1}(\boldsymbol{y} - X\hat{\boldsymbol{\beta}}) \\ &= (\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})(X^\top V^{-1}\boldsymbol{y} - (X^\top V^{-1} X)\hat{\boldsymbol{\beta}}) \\ &= 0\end{aligned}$$

where

$$\hat{\boldsymbol{\beta}} = (X^T V^{-1} X)^{-1} X^T V^{-1} \boldsymbol{y}, \tag{2.25}$$

the generalised least squares estimator of $\boldsymbol{\beta}$. We use the same trick of making the integrand 'look like' a multivariate normal density function, so that the integral will equal 1. We write (with p equal to the number of elements in $\boldsymbol{\beta}$)

$$L_R(\boldsymbol{\theta}, \sigma^2|\boldsymbol{y}) = \frac{|V|^{-\frac{1}{2}}}{(2\pi)^{\frac{n}{2}}} \exp\left\{-\frac{1}{2}(\boldsymbol{y} - X\hat{\boldsymbol{\beta}})^T V^{-1}(\boldsymbol{y} - X\hat{\boldsymbol{\beta}})\right\} \tag{2.26}$$

$$\times \frac{|(X^T V^{-1} X)^{-1}|^{\frac{1}{2}}}{(2\pi)^{-\frac{p}{2}}} \int \frac{|(X^T V^{-1} X)^{-1}|^{-\frac{1}{2}}}{(2\pi)^{\frac{p}{2}}} \exp\left\{-\frac{1}{2}(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})^T X^T V^{-1} X(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}))\right\} d\boldsymbol{\beta}$$

$$= \frac{|V|^{-\frac{1}{2}}}{(2\pi)^{\frac{n}{2}}} \exp\left\{-\frac{1}{2}(\boldsymbol{y} - X\hat{\boldsymbol{\beta}})^T V^{-1}(\boldsymbol{y} - X\hat{\boldsymbol{\beta}})\right\} \frac{|X^T V^{-1} X|^{-\frac{1}{2}}}{(2\pi)^{-\frac{p}{2}}} \times 1, \tag{2.27}$$

as the integrand is the density function of a normal random variable ($\boldsymbol{\beta}$) with mean $\hat{\boldsymbol{\beta}}$ and variance $(X^T V^{-1} X)^{-1}$. This gives us the REML criterion. R will maximise this as a function of σ^2 and $\boldsymbol{\theta}$, when estimating the parameters using REML. The fixed effects are then estimated using (2.25), with the estimates of σ^2 and $\boldsymbol{\theta}$ used to construct V.

Note that the value of $|X^T V^{-1} X|$ can change for two identical models using different parameterisation (e.g. constraints on the fixed effects parameters). We discuss the significance of this later when we come to model comparisons.

Summary

- The default method in `lme4`, REML, is to first estimate the variance parameters $\boldsymbol{\theta}$ by maximising (2.27), and then estimate the fixed effects $\boldsymbol{\beta}$ using (2.25), with V calculated using the estimated value of $\boldsymbol{\theta}$.
- Ordinary maximum likelihood estimates $\boldsymbol{\theta}$ and $\boldsymbol{\beta}$ simultaneously by maximising (2.23). This can be done in `lme4` by including the argument `REML=F`.
- The distinction between REML and ML is less important than is often claimed. For fixed effects models, REML gives unbiased estimates of σ^2. But for mixed effects models, REML and ML give biased estimates. Moreover, it is not clear why we should care that our estimators are unbiased for σ^2 - they will still be biased estimators of σ, which is usually of greater interest.

  REML seems to have been historically preferred because it gives the same variance estimates as the classical estimators for balanced designs.
- The maximised value of (2.27) will change for two different, but equivalent parameterisations of the fixed effects. This means that for hypothesis testing, we **must** use maximum likelihood, not REML.

Task 5. *Work through an implementation of these methods on the rail data (without using the `lmer` command), using the R script `MAS474-REML.R`. Note that the implementation of REML in `lme4` is considerably more sophisticated, using various numerical and linear algebra techniques to speed up the computation.*

Likelihood values in R

Having fitted the model with `lmer`, you can obtain the maximised REML or ordinary (log) likelihood value using the `logLik` command. R will use the same likelihood (REML or ordinary) used to fit the model. If the model has been fitted with REML, the `summary` command (and viewing the model fit) will produce the output (for the rail data)

```
REML criterion at convergence: 122.177
```

with the value reported being -2 times the maximised log of (2.27).

Task 6. *Investigate the effects of different fixed effects parameterisations in the* `ergoStool` *data. Try sum-to-zero contrasts by include the argument*

```
contrasts=list(Type=contr.sum)
```

Compare the fits under the two options. You should find that estimates of the variance parameters do not change, but the log-restricted likelihood values do.

2.8 Predicting random effects: best linear unbiased predictions

In a mixed effects model, we don't actually estimate the random effects, rather, we estimate the corresponding variance parameters. Given the estimated variance parameter (and observed data), we predict the random effects. (We distinguish between *estimating* parameters, and *predicting* random variable).
As before, we write the model as

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{Z}\boldsymbol{b} + \boldsymbol{\epsilon}.$$

and suppose that the random effects $\boldsymbol{b}$ are modelled as a normal random vector with mean $\mathbf{0}$, and that $\boldsymbol{b}$ is independent of $\boldsymbol{\epsilon}$, also normal with mean $\mathbf{0}$.

Given observations $\boldsymbol{Y}$, the minimum mean square error predictor of $\boldsymbol{b}$ can be shown to be the conditional expectation[5]

$$\widehat{\boldsymbol{b}} = E(\boldsymbol{b}|\boldsymbol{Y}).$$

The normal assumptions allow us to find $E(\boldsymbol{b}|\boldsymbol{Y})$ more explicitly. To do so, we will need the following result about conditioning multivariate normals.

Theorem 1. *If n dimensional $\boldsymbol{y}$ is partitioned as $\boldsymbol{y} = (\boldsymbol{y}_1, \boldsymbol{y}_2)^\top$, and*

$$\begin{pmatrix} \boldsymbol{y}_1 \\ \boldsymbol{y}_2 \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} \boldsymbol{m}_1 \\ \boldsymbol{m}_2 \end{pmatrix}, \begin{pmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{pmatrix} \right)$$

then

$$\boldsymbol{y}_2 \mid \boldsymbol{y}_1 \sim \mathcal{N}\left(\boldsymbol{m}_2 + V_{21}V_{11}^{-1}(\boldsymbol{y}_1 - \boldsymbol{m}_1), V_{22} - V_{21}V_{11}^{-1}V_{12}\right)$$

[5] Another rationale for using the posterior mean comes from decision theory. If we use a quadratic loss function, the best estimator of a quantity (in the sense of minimizing the loss) is given by the posterior mean.

Proof. Let $\boldsymbol{z} = \boldsymbol{y}_2 + A\boldsymbol{y}_1$ where $A = -V_{21}V_{11}^{-1}$. Then

$$\begin{aligned}\mathbb{C}\mathrm{ov}(\boldsymbol{z}, \boldsymbol{y}_1) &= \mathbb{C}\mathrm{ov}(\boldsymbol{y}_2, \boldsymbol{y}_1) + \mathbb{C}\mathrm{ov}(A\boldsymbol{y}_1, \boldsymbol{y}_1) \\ &= V_{21} - V_{21} \\ &= 0\end{aligned}$$

so $\boldsymbol{z}$ and $\boldsymbol{y}_1$ are uncorrelated. Since they are jointly normal (as linear combinations of normals are still normal), they are also independent.

Clearly, $\mathbb{E}\boldsymbol{z} = \boldsymbol{m}_2 + A\boldsymbol{m}_1$, and thus

$$\begin{aligned}\mathbb{E}(\boldsymbol{y}_2 \mid \boldsymbol{y}_1) &= \mathbb{E}(\boldsymbol{z} - A\boldsymbol{y}_1 \mid \boldsymbol{y}_1) \\ &= \mathbb{E}(\boldsymbol{z} \mid \boldsymbol{y}_1) - A\boldsymbol{y}_1 \\ &= \boldsymbol{m}_2 + V_{21}V_{11}^{-1}(\boldsymbol{y}_1 - \boldsymbol{m}_1)\end{aligned}$$

as required. For the covariance matrix,

$$\begin{aligned}\mathbb{V}\mathrm{ar}(\boldsymbol{y}_2|\boldsymbol{y}_1) &= \mathbb{V}\mathrm{ar}(\boldsymbol{z} - A\boldsymbol{y}_1|\boldsymbol{y}_1) \\ &= \mathbb{V}\mathrm{ar}(\boldsymbol{z}|\boldsymbol{y}_1) \\ &= \mathbb{V}\mathrm{ar}(\boldsymbol{z}) \\ &= \mathbb{V}\mathrm{ar}(\boldsymbol{y}_2 + A\boldsymbol{y}_1) \\ &= \mathbb{V}\mathrm{ar}(\boldsymbol{y}_2) + A\mathbb{V}\mathrm{ar}(\boldsymbol{y}_1)A^\top + A\mathbb{C}\mathrm{ov}(\boldsymbol{y}_2, \boldsymbol{y}_1) + \mathbb{C}\mathrm{ov}(\boldsymbol{y}_1, \boldsymbol{y}_2)A^\top \\ &= V_{22} - V_{21}V_{11}^{-1}V_{12}\end{aligned}$$

as required. ■

If we write the joint normal distribution of $\boldsymbol{Y}$ and $\boldsymbol{b}$ as

$$\begin{pmatrix}\boldsymbol{Y} \\ \boldsymbol{b}\end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix}\boldsymbol{X}\boldsymbol{\beta} \\ \boldsymbol{0}\end{pmatrix}, \begin{pmatrix}V_Y & C_{Y,b} \\ C_{b,Y} & V_b\end{pmatrix}\right),$$

where $V_Y, V_b, C_{Y,b}$ and $C_{b,Y}$ are the appropriate variance and covariance matrices, then the theorem tells us that the conditional distribution of $\boldsymbol{b}$ given $\boldsymbol{Y}$ is the Normal distribution:

$$\boldsymbol{b}|\boldsymbol{Y} \sim \mathcal{N}\left(C_{b,Y}V_Y^{-1}(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}), V_b - C_{b,Y}V_Y^{-1}C_{Y,b}\right).$$

Thus

$$\widehat{\boldsymbol{b}} = E(\boldsymbol{b}|\boldsymbol{Y}) = C_{b,Y}V_Y^{-1}(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}).$$

This depends on the unknown $\boldsymbol{\beta}$. Substituting $\widehat{\boldsymbol{\beta}}$ gives the BLUP:

$$\mathrm{BLUP}(\boldsymbol{b}) = \widehat{\boldsymbol{b}}(\widehat{\boldsymbol{\beta}}) = C_{b,Y}V_Y^{-1}(\boldsymbol{Y} - \boldsymbol{X}\widehat{\boldsymbol{\beta}}). \tag{2.28}$$

It can be shown that amongst linear functions, $\widetilde{\boldsymbol{b}}$ say, of $\boldsymbol{Y}$ that are unbiased for $\boldsymbol{b}$ in the sense that $E(\widetilde{\boldsymbol{b}}) = E(\boldsymbol{b})$, the BLUP as defined here is the one that minimizes $E(\widetilde{\boldsymbol{b}} - \boldsymbol{b})'A(\widetilde{\boldsymbol{b}} - \boldsymbol{b})$ for any positive definite symmetric matrix A of appropriate dimension.

The BLUP is also dependent on the unknown $\boldsymbol{\theta}$ (which includes the unknown variance parameters), so in practice, we actually use an estimated BLUP:

$$\widehat{\text{BLUP}}(\boldsymbol{b}) = \widehat{\boldsymbol{b}}(\widehat{\boldsymbol{\beta}}, \widehat{\boldsymbol{\theta}}) = \widehat{C}_{b,Y}\widehat{V}_Y^{-1}(\boldsymbol{Y} - \boldsymbol{X}\widehat{\boldsymbol{\beta}}), \tag{2.29}$$

where $\widehat{C}_{b,Y}$ and $\widehat{V}_Y$ are calculated using the estimated value of $\boldsymbol{\theta}$.

2.8.1 Obtaining the predicted random effects in R

These can be obtained with the `ranef` command. For the rail-data example, we have

```
> fm1<-lmer(travel~1+(1|Rail),raildata)
> ranef(fm1)
$Rail
  (Intercept)
1   -12.39148
2   -34.53091
3    18.00894
4    29.24388
5   -16.35675
6    16.02631
```

so $\hat{b}_1 = -12.39$ (to 2 d.p.) and so on. The R script shows the calculation of these predicted random effects for the rail data, by constructing the terms required in (2.29).

Note that to get the posterior variances, $\mathbb{V}\text{ar}(\boldsymbol{b}|\boldsymbol{Y})$, we need to do the calculation ourselves as this is not currently implemented in `lmer`.

2.9 Comparing Mixed and Fixed Effects Models

The decision to use fixed or random effects is a *modelling choice*, determined by what we are trying to describe and infer. It does not make sense to *test* whether a grouping variable should be included as a fixed effect or a random effect. We should, however, test the model assumptions, and we discuss doing so in a later chapter.

Nevertheless, it is important to understand the consequences of switching between fixed and random effects. We have already discussed this in 2.3, but here we give another example with the rail data. We first fit the following two models in R.

```
> lm.fixed<-lm(travel~Rail,contrasts=list(Rail=contr.sum),raildata)
> lm.mixed<-lmer(travel~1+(1|Rail),raildata)
```

We write the fixed effects model as

$$\begin{aligned} y_{ij} &= \beta + \alpha_i + \epsilon_{ij}, \quad i = 1, ..., 6, \quad j = 1, ..., 3, \\ & \sum_{i=1}^{6} \alpha_i = 0, \quad \epsilon_{ij} \sim N(0, \sigma^2). \end{aligned} \tag{2.30}$$

Inspecting the coefficients and standard errors, both models give the same estimate value for β: 66.5. However, the standard error is larger in `lm.mixed` (10.17 compared to 0.95) . This is because, in this model β is the mean of the expected travel times for the 'population' of rails, from which we have a sample of six. Given only six rails, we shouldn't expect to know β with certainty. In `lm.fixed`, β is the mean of the expected travel times *for the six rails in the study.* As there is relatively little variation within rails, we have relatively little uncertainty about β.

The least squares estimates $\hat{\alpha}_i$ are similar, but not identical, to the predicted random effects $\hat{b}_i$. In general, when comparing the two, there will be some 'shrinkage' from the least squares estimates towards zero, as a consequence of modelling the random effects as random variables with expectation zero. The amount of shrinkage increases as σ_b^2 decreases relative to σ^2.

Task 7. *Compare the mixed effects model for the* `ergoStool` *data fitted previously with a fixed effects version obtained with the* `lm` *function*
(with model formula `effort ~ Subject + Type, contrasts=list(Subject=contr.sum)`*).*

Chapter 3

Mixed Effects Models: Further Examples

3.1 Multilevel Models

So far we have only considered one level of grouping in the data, but our data are often more highly structured. For example we might have a **nested arrangement** of pupils within classes within schools or a **split-plot** agricultural experiment with crop varieties applied to different plots within a field and then different fertilizer or harvesting regimes applied to different parts (subplots) of each plot.

3.1.1 A Nested Example

The data set `Oxide` demonstrates the new features. The data arise from an observational study in the semiconductor industry. The response variable is the thickness of oxide coating on silicon wafers measured at three different sites on each of three wafers selected from each of eight lots sampled from the population of possible lots.

The objective of the study was to estimate the variance components in order to determine assignable causes of observed variability and so we have no fixed effects of interest, other than the overall mean response. Thus we can concentrate on the grouping structure of the experimental units and the way to represent this with random effects. Each site is specific to each wafer with, for example, no link between site 1 on wafer 1 and site 1 on any other wafer. We speak of `Site` being 'nested' within `Wafer`, or simply the effect of `Site` within `Wafer`. Similarly, the wafers are nested within the lots. We will expect our initial ideas of a hi-

erarchy of successively weaker correlations to apply, in the sense that measurements at sites on a single wafer will be expected to be more similar than those on different wafers, which in turn will be more similar than those between lots. We model this by adding random effects for each level:

$$\begin{aligned} Y_{ijk} &= \beta + b_i + b_{ij} + \epsilon_{ijk}, \quad i = 1, ..., 8, \quad j, k = 1, 2, 3, \\ & b_i \sim N(0, \sigma_1^2), \quad b_{ij} \sim N(0, \sigma_2^2), \quad \epsilon_{ijk} \sim N(0, \sigma^2) \end{aligned} \tag{3.1}$$

where Y_{ijk} represents the oxide thickness for the kth site on the jth wafer from the ith lot, β is the mean thickness (fixed effect), b_i is the random effect of the ith lot, b_{ij} is the random effect of the jth wafer from the ith lot and ϵ_{ijk} is the random error for the experimental units. Note that there is no b_j term as there is no relationship between the j^{th} wafers in different lots.

A special notation is used to specify nested grouping structures: for factors `G1, G2,...,Gk` the notation `G1/G2/.../Gk` means `Gk` within `G(k-1)`,...,`G3` within `G2`, `G2` within `G1`. We use this to specify the random part of the model (3.1) in a call to `lmer`.

The model (3.1) is fitted simply by

```
lmer(Thickness~1+(1|Lot/Wafer),data=Oxide)
```

since the only fixed effect in (3.1) is the single constant term.

```
> fm1<-lmer(Thickness~1+(1|Lot/Wafer),data=Oxide)
> summary(fm1)
Linear mixed model fit by REML ['lmerMod']
Formula: Thickness ~ 1 + (1 | Lot/Wafer)
   Data: Oxide

REML criterion at convergence: 454

Scaled residuals:
    Min      1Q  Median      3Q     Max
-1.8746 -0.4991  0.1047  0.5510  1.7922

Random effects:
 Groups    Name        Variance Std.Dev.
 Wafer:Lot (Intercept)  35.87    5.989
 Lot       (Intercept) 129.91   11.398
 Residual               12.57    3.545
Number of obs: 72, groups:  Wafer:Lot, 24; Lot, 8

Fixed effects:
            Estimate Std. Error t value
(Intercept) 2000.153      4.232   472.7
```

We see that $\widehat{\sigma}_1 = 11.40$, $\widehat{\sigma}_2 = 5.99$ and $\widehat{\sigma} = 3.54$.

N0te that

```
lmer(Thickness~1 +(1|Lot)+(1|Lot:Wafer), data =Oxide)
```

fits the same model.

3.1.2 A Split-Plot Example

The basic formulation of a split-plot model is similar to that of the nested model in Section 3.1.1, except that almost always we will have genuine experimental factors which need to be accounted for with fixed effects and also we may encounter a conventional alternative way of expressing the grouping structure.

The data set `Oats` illustrates these possibilities. The treatment structure here was a full 3×4 factorial arrangement with three varieties of oats and four concentrations of nitrogen as a fertilizer. The agricultural land available for the experiment was divided into six blocks and each of these was divided into three plots to which the varieties were randomly assigned (one per plot within each block). Each plot was then subdivided into four subplots and the nitrogen concentrations assigned randomly one per subplot within each plot. Thus we have a grouping structure of subplots within plots within blocks and again a natural hierarchy of correlations might be anticipated. Once again, with only a single observation per subplot, we should not include an actual subplot effect.

A suitable model would be

$$Y_{ijk} = \mu + \tau_{v(i,j)} + \beta x_{ijk} + b_i + b_{ij} + \epsilon_{ijk}, \qquad (3.2)$$

with Y_{ijk} the yield in block i, plot j and subplot k. The term $v(i,j)$ gives the variety of oats used in block i, plot j, and x_{ijk} is the nitro level in block i, plot j and subplot k. We assume

$$b_i \sim N(0, \sigma_1^2), \quad b_{ij} \sim N(0, \sigma_2^2), \quad \epsilon_{ijk} \sim N(0, \sigma^2).$$

In R, we have

```
> fm1<-lmer(yield~nitro+Variety+(1|Block/Variety),Oats)
> summary(fm1)
Linear mixed model fit by REML ['lmerMod']
Formula: yield ~ nitro + Variety + (1 | Block/Variety)
   Data: Oats
```

```
REML criterion at convergence: 578.9

Scaled residuals:
     Min       1Q   Median       3Q      Max
-1.62948 -0.65841 -0.07207  0.55785  1.71463

Random effects:
 Groups        Name        Variance Std.Dev.
 Variety:Block (Intercept) 108.9    10.44
 Block         (Intercept) 214.5    14.65
 Residual                  165.6    12.87
Number of obs: 72, groups:  Variety:Block, 18; Block, 6

Fixed effects:
                  Estimate Std. Error t value
(Intercept)         82.400      8.059  10.225
nitro               73.667      6.781  10.863
VarietyMarvellous    5.292      7.079   0.748
VarietyVictory      -6.875      7.079  -0.971

Correlation of Fixed Effects:
            (Intr) nitro  VrtyMr
nitro       -0.252
VartyMrvlls -0.439  0.000
VarityVctry -0.439  0.000  0.500
```

We see that $\widehat{\sigma}_1 = 14.65$, $\widehat{\sigma}_2 = 10.44$ and $\widehat{\sigma} = 12.87$.

This type of experimental structure is particularly common in agricultural experiments. The division of available land into blocks will usually be done so that blocks are relatively homogeneous, but this is not essential for the use of a mixed model. Here we simply need to know that the grouping structure is block $\supset$ plot $\supset$ subplot.

3.2 Random Interaction Terms

An interaction between a random and a fixed effect may be needed for realistic modelling. For example, in the data set `Machines` we have three replicate productivity scores for each of six randomly selected workers on three machines, and the plot of each worker's average score on each machine shown in Figure 3.1 suggests that there may indeed be differences between workers in how well they perform on the different machines; that

is `Machine:Worker` interactions.

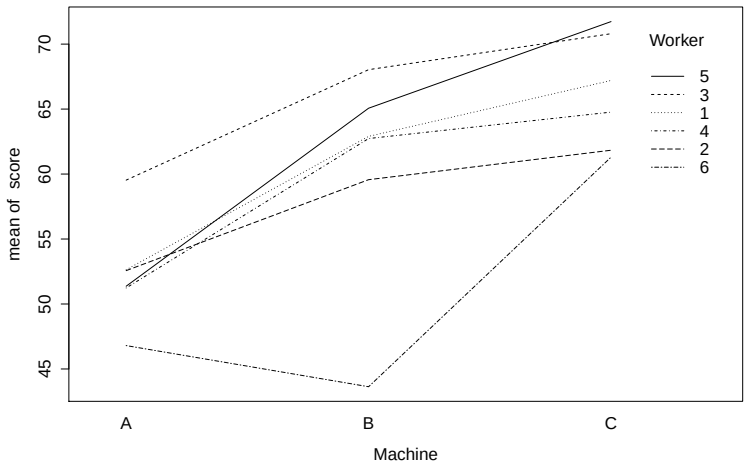

Figure 3.1: Interaction plot for productivity scores

The plot in Figure 3.1 is an *interaction plot* produced by the function `interaction.plot`:

```
interaction.plot(Machines$Machine, Machines$Worker, Machines$score)
```

Because the workers represent a random sample from the population of workers, it is natural to treat the `Worker` factor as a random effect. On the other hand, interest in the particular machines suggests that the `Machines` factor should be a fixed effect. A simple additive model

$$Y_{ijk} = \beta_j + b_i + \epsilon_{ijk} \tag{3.3}$$

in which Y_{ijk} represents the kth replicate productivity score for worker i on machine j and b_i is a random effect for worker i and β_j a fixed effect for machine j would produce approximately parallel lines in the interaction plot. An interaction term between `Worker` and `Machine` is clearly called for.

Since the workers are a random sample, interaction terms modelling the pattern of their productivity from one machine to another will be expressed as random effects. The appropriate model is obtained by simply adding random term b_{ij} to the right hand side of (3.3). There are then two levels of random effects: the b_i for workers, and the b_{ij} for the type of machine within each worker. In the `lmer` function we can therefore express the random part simply by specifying a `Worker/Machine` nested structure, meaning '`Worker` then `Machine`-within-`Worker`'.

```
> fm1<-lmer(score~Machine-1+(1|Worker/Machine),data=Machines)
```

```
> summary(fm1)
Linear mixed model fit by REML ['lmerMod']
Formula: score ~ Machine - 1 + (1 | Worker/Machine)
   Data: Machines

REML criterion at convergence: 215.7

Scaled residuals:
     Min       1Q   Median       3Q      Max
-2.26959 -0.54847 -0.01071  0.43937  2.54006

Random effects:
 Groups         Name        Variance Std.Dev.
 Machine:Worker (Intercept) 13.9095  3.7295
 Worker         (Intercept) 22.8584  4.7811
 Residual                    0.9246  0.9616
Number of obs: 54, groups:  Machine:Worker, 18; Worker, 6

Fixed effects:
         Estimate Std. Error t value
MachineA   52.356      2.486   21.06
MachineB   60.322      2.486   24.27
MachineC   66.272      2.486   26.66

Correlation of Fixed Effects:
         MachnA MachnB
MachineB 0.617
MachineC 0.617  0.617
```

3.2.1 Understanding the covariance structures

Consider the previous example and the model

$$Y_{ijk} = \beta_j + b_i + b_{ij} + \epsilon_{ijk}, \tag{3.4}$$

with $b_i \sim N(0, \sigma_1^2)$, $b_{ij} \sim N(0, \sigma_2^2)$, and $\epsilon_{ijk} \sim N(0, \sigma^2)$, where i represents worker, j represents machine, and k represents replicate. The command

```
(fm1<-lmer(score~Machine-1+(1|Worker/Machine),data=Machines))
```

forces all the random effects to be independent. Observations between different workers are independent, but observations on the same worker are correlated. If we consider the observations for worker i, we have, for $k \neq k'$ and $j \neq j'$,

$$
\begin{aligned}
Var(Y_{ijk}) &= \sigma_1^2 + \sigma_2^2 + \sigma^2 && (3.5)\\
Cov(Y_{ijk}, Y_{ijk'}) &= Cov(\beta_j + b_i + b_{ij} + \epsilon_{ijk}, \beta_j + b_i + b_{ij} + \epsilon_{ijk'}) && (3.6)\\
&= Cov(b_i + b_{ij}, +b_i + b_{ij}) && (3.7)\\
&= \sigma_1^2 + \sigma_2^2 && (3.8)\\
Cov(Y_{ijk}, Y_{ij'k'}) &= Cov(\beta_j + b_i + b_{ij} + \epsilon_{ijk}, \beta_j + b_i + b_{ij'} + \epsilon_{ij'k'}) && (3.9)\\
&= Cov(b_i, b_i) && (3.10)\\
&= \sigma_1^2. && (3.11)
\end{aligned}
$$

From the R output

```
Random effects:
 Groups         Name        Variance Std.Dev.
 Machine:Worker (Intercept) 13.90945 3.72954
 Worker         (Intercept) 22.85849 4.78105
 Residual                    0.92463 0.96158
```

we get

$$\hat{\sigma}_1^2 = 22.85849, \qquad \hat{\sigma}_2^2 = 13.90945, \qquad \hat{\sigma}^2 = 0.92463. \tag{3.12}$$

Correlated random effects

The command

```
(fm3<-lmer(score~Machine-1+(Machine-1|Worker),data=Machines))
```

fits a model that appears, at first glance, to be simpler:

$$Y_{ijk} = \beta_j + b_{ij} + \epsilon_{ijk}. \tag{3.13}$$

However, we now have

$$\mathbf{b} = \begin{pmatrix} b_{i1} \\ b_{i2} \\ b_{i3} \end{pmatrix} \sim MVN(\mathbf{0}, \Sigma)$$

where

$$\Sigma_{mn} = \begin{pmatrix} \sigma_1^2 & \rho_{12}\sigma_1\sigma_2 & \rho_{13}\sigma_1, \sigma_3 \\ \rho_{12}\sigma_1\sigma_2 & \sigma_2^2 & \rho_{23}\sigma_2, \sigma_3 \\ \rho_{13}\sigma_1\sigma_3 & \rho_{23}\sigma_2, \sigma_3 & \sigma_3^2 \end{pmatrix}$$

That is

$$
\begin{aligned}
Var(b_{ij}) &= \sigma_j^2, && (3.14)\\
Cov(b_{ij}, b_{ij'}) &= \rho_{j,j'}\sigma_j\sigma_{j'}. && (3.15)
\end{aligned}
$$

From the R output

```
Random effects:
 Groups   Name     Variance Std.Dev. Corr
 Worker   MachineA 16.64049 4.07928
          MachineB 74.39527 8.62527  0.803
          MachineC 19.26754 4.38948  0.623 0.771
 Residual           0.92463 0.96158
```

we get

$$\hat{\sigma}_1^2 = 16.64049, \qquad \hat{\sigma}_2^2 = 74.39527, \qquad \hat{\sigma}_3^2 = 19.26754, \qquad \hat{\sigma}^2 = 0.92463 \tag{3.16}$$

$$\hat{\rho}_{12} = 0.803, \qquad \hat{\rho}_{13} = 0.623, \qquad \hat{\rho}_{23} = 0.771. \tag{3.17}$$

We have a more flexible covariance structure

$$\begin{aligned} Var(Y_{ijk}) &= \sigma_j^2 + \sigma^2 && (3.18)\\ Cov(Y_{ijk}, Y_{ijk'}) &= Cov(\beta_j + b_{ij} + \epsilon_{ijk}, \beta_j + b_{ij} + \epsilon_{ijk'}) && (3.19)\\ &= Cov(b_{ij}, b_{ij}) && (3.20)\\ &= \sigma_j^2 && (3.21)\\ Cov(Y_{ijk}, Y_{ij'k'}) &= Cov(\beta_j + b_{ij} + \epsilon_{ijk}, b_{ij'} + \epsilon_{ij'k'}) && (3.22)\\ &= Cov(b_{ij}, b_{ij'}) && (3.23)\\ &= \rho_{j,j'}\sigma_j\sigma_{j'}. && (3.24) \end{aligned}$$

Comparing the variance and covariance formulae for the two models, we can see the following.
For `fm1`:

- all observations have the same variance;
- the covariance between observations corresponding to the same worker using different machines is the same, for any pair of machines.

For `fm3`:

- the variance of an observation depends on the machine being used;
- the covariance between observations corresponding to the same worker using different machines is different, for different pairs of machines.

We could introduce even more flexibility by allowing the error variance to change between the machines, but this is harder to implement using `lme4`. This is the benefit of the Bayesian approach - it allows us to specify more general models.

3.3 Repeated Measures

In all examples so far the fixed effect part of the statistical model has been determined by factors, either singly or in combination. We now consider situations in which the the fixed effects are covariates, and so we are now in the realms of generalizing linear regression models (or analyses of covariance – where both covariates and factor effects appear) to include random effects. Typical examples are growth curves and other forms of repeated measures data.

The example we shall use is the data set `Orthodont`, though we shall concentrate on only the female subjects' data. The data consist of measurement of the distance between the pituitary gland and the pterygomaxillary fissure (`distance`)– two easily-identified points on X-radiographs of the skull – on each of 16 male and 11 female children at four times (at ages 8, 10, 12 and 14). The main aim is to model growth curves for the children by quantifying the variation of the distance with age. Accordingly the basic fixed effects structure is taken to be a regression of `distance` on `age`. A conditioning plot of the data (a plot of `distance` against `age` for each `Subject` shows that the distance-age relationship varies somewhat for males and females, so we concentrate here on the female data only.

To obtain the conditioning plot, type

```
plot(Orthodont)
```

and to obtain the same plot for only the female subjects, first use subsetting to extract the relevant data:

```
plot(Orthodont[ Orthodont$Sex == "Female", ] )
```

Recognizing that the data are collected in groups of four observations per subject, we may wish to include a random effects term to allow for within-subject correlation. If we just include

```
random = ~ 1 | Subject
```

in the argument for `lme` we will fit the model

$$\begin{aligned} Y_{ij} &= \beta_1 + b_i + \beta_2 x_{ij} + \epsilon_{ij}, \quad i = 1, ..., M, \quad j = 1, ..., n_i, \\ & \qquad b_i \sim N(0, \sigma_b^2), \quad \epsilon_{ij} \sim N(0, \sigma^2), \end{aligned} \tag{3.25}$$

where y_{ij} is the distance between the points for the ith subject at the jth occasion, x_{ij} is the age of the ith subject at occasion j, β_1 is the intercept fixed effect, β_2 is the slope fixed effect, b_i is the random effect of the ith subject, ϵ_{ij} is the within-subject random error, M is the number of subjects (here 11) and n_i is the number of observations made on subject i (here 4 for all subjects).

Task 8. *Fit this model to the* `Orthodont` *data. The restriction to female subjects only may be made by including* `subset = Sex == "Female"` *as an extra argument of the* `lmer` *function.*

Including a random intercept term in the model, as in (3.25), will allow us to model the substantial differences in general distance for the different girls seen in our original plot; for example, subject 11's measurements are generally smaller than those of subject 10. However, closer examination of the multipanel plot also shows appreciable variation in slope between the girls; compare subjects 2 and 8, for example. The plot suggests that girls are growing at different rates. In order to model this feature we need to introduce a random effect which is an additive modification for each subject to the **slope** parameter of the underlying regression – a different construction from any we have used before. Assuming that we still include a random effect on the intercept, we are now envisaging the model

$$\begin{aligned} Y_{ij} &= \beta_1 + b_{1i} + (\beta_2 + b_{2i})x_{ij} + \epsilon_{ij}, \quad i = 1, ..., M, \quad j = 1, ..., n_i, \\ & \qquad b_{1i} \sim N(0, \sigma_1^2), \quad b_{2i} \sim N(0, \sigma_2^2), \quad \epsilon_{ij} \sim N(0, \sigma^2). \end{aligned} \tag{3.26}$$

An interpretation of model (3.26) is that for each subject there is a straight line relationship between distance and age. The lines for different subjects are basically similar, but differ slightly (assuming σ_1^2 and σ_2^2 are small) in slope and intercept. Estimates of the fixed effects parameters β_1 and β_2 tell us about average characteristics of the population from which the subjects were drawn, and estimates of the variances σ_1^2 and σ_2^2 tell us about variability amongst subjects. This random effects model is more parsimonious than one with different fixed slopes and intercepts for each subject and its interpretation may be preferable.

To fit the model, simply enter `age | Subject` as the random part of the model in `lmer`. (The convention here is that on the right hand side of the model formula a constant term is understood; thus `~ age` is shorthand for `~ 1 + age`. This is the same convention for model formulae as elsewhere in R.)

As it stands, the model expressed by (3.26) is incomplete since it does not specify the relationship between the random effects b_{1i} and b_{2i}. By default the `lmer` function takes them to be independent from group to group but within a group allows them to be arbitrarily correlated; estimates of their correlation matrix are returned within the fitted `lmer` object.

Task 9. *Fit model 3.26 in* `R` *and note the estimated correlation between random effects.*

Chapter 4

Mixed Effects Models: Model Checking and Inference

4.1 Checking the model assumptions

As usual we assess the fitted model by examining numerical and graphical summaries. Here we check that there are no gross departures from assumptions, that the model explains the data adequately and that it is not unnecessarily complex. We return to the `ergostool` dataset. To recap, we have the model

$$Y_{ij} = \beta_j + b_i + \epsilon_{ij}, \quad i = 1, \ldots, 9, \quad j = 1, \ldots, 4, \quad (4.1)$$

where Y_{ij} represents the effort required by the ith subject to rise from the jth type of stool, β_j is the fixed effect of the jth type of stool, b_i is the random effect of the ith subject and ϵ_{ij} is the random error for the experimental units. We suppose that

$$b_i \sim N(0, \sigma_b^2), \quad \epsilon_{ij} \sim N(0, \sigma^2). \quad (4.2)$$

The assumptions made in the present model are therefore

- Assumption 1: the random effects b_i are i.i.d. $N(0, \sigma_b^2)$.
- Assumption 2: the within-group (residual) errors ϵ_{ij} are i.i.d. $N(0, \sigma^2)$, and are independent of the random effects.
- Assumption 3: the fixed effects β_j adequately represent the mean response.

To assess the validity of Assumption 1 we use estimates (predictors) of the random effects b_i, the Best Linear Unbiased Predictors $\hat{b}_i$. These are estimates of the conditional expectations of

the b_i given the observed data $\{y_{ij}\}$, as discussed in 2.8. They are calculated by the `ranef` function applied to the `lmer` object:

```
> fm1<-lmer(effort~Type + (1|Subject),ergoStool)
> ranef(fm1)
$Subject
    (Intercept)
1  1.708781e+00
2  1.708781e+00
3  4.271951e-01
4 -8.543903e-01
5 -1.495183e+00
6  1.290500e-14
7  4.271951e-01
8 -1.708781e+00
9 -2.135976e-01
```

A dot plot of the $\hat{b}_i$ may be useful in highlighting outliers or unusual `Subjects`. If there is no evidence of outliers or other non-stationarity, we use a Normal QQ Plot to assess Normality.

The dot plot is obtained by

```
plot(ranef(fm1)$Subject)
```

To obtain the Normal QQ plot we can use `qqnorm`. Because the output from `ranef` is actually a *list* rather than a vector, we first `unlist` `ranef`. A reference line with slope $\widehat{\sigma}_b$ is appropriate and is added with the `abline` argument:

```
qqnorm(unlist(ranef(erg1.lme)))
abline(0,1.332)
```

The plot is shown in Figure 4.1. In the present example the plot gives no reason to question normality of the b_i.

To assess Assumptions 2 and 3 we use fitted values and residuals. In standard linear models these would be $\widehat{E(Y)}$ and $\hat{\epsilon} = y - \widehat{E(Y)}$ respectively. Here the more complex random structure requires a more refined approach. At the level of an individual `Subject`, i say, it is reasonable to take the fitted values to be $\hat{b}_i + \hat{\beta}_j$, but at the population level, that is, averaged over `Subjects`, the fitted values would be simply $\hat{\beta}_j$. Thus we now have two sets of fitted values. The subject-wise fitted values are said to be *at the inner level* or *level 1*, and the population fitted values *at the outer level* or *level 0*.

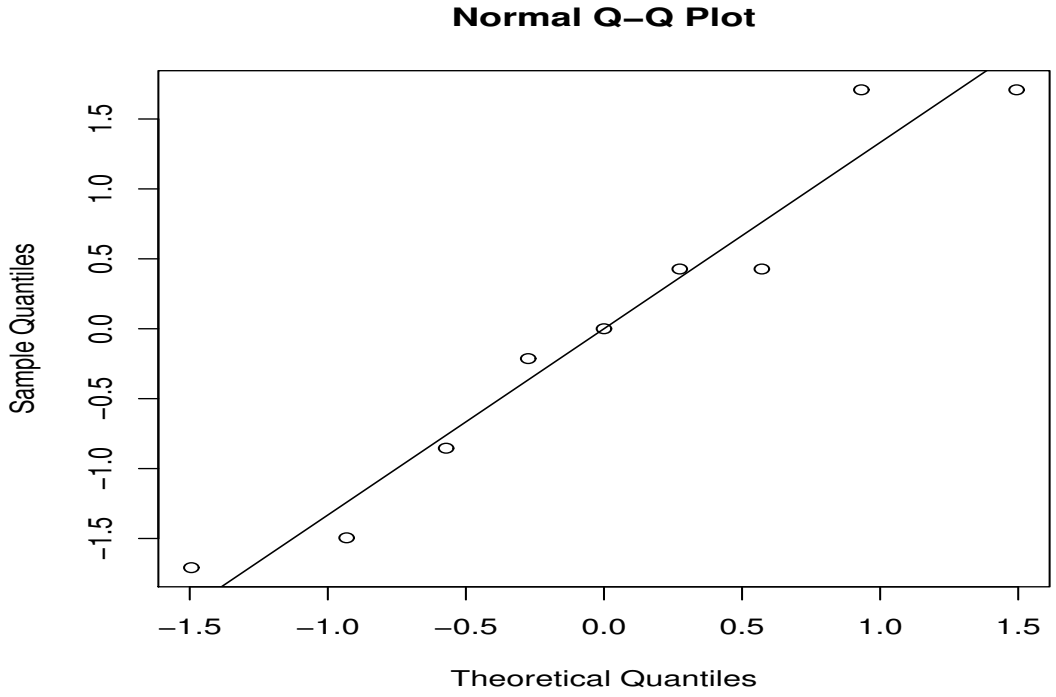

Figure 4.1: QQnorm plot for the Ergo data

Fitted Values

Name(s)		Level	Fitted values
Subject-level	≡ *Within-group* ≡ *Inner-level*	1	$\hat{b}_i + \hat{\beta}_j$
Population-level	≡ *Outer-level*	0	$\hat{\beta}_j$

Correspondingly there are two levels of residuals: the *within-group residuals*, the estimates of the ϵ_{ij} defined as the difference between the observed values and the within-group fitted values, also called the *subject-level, inner level* or *level 1* residuals, and the *population-level residuals*, the usual difference between the observed values and their estimated means, also called the *outer-level* or *level 0* residuals (mnemonic: **O**uter = level **0**, the crudest level).

Residuals

Name(s)		Level	Residuals
Subject-level	≡ *Within-group* ≡ *Inner-level*	1	$\hat{\epsilon}_{ij} = y_{ij} - \hat{b}_i - \hat{\beta}_j$
Population-level	≡ *Outer-level*	0	$\hat{\epsilon}_{ij} + \hat{b}_i = y_{ij} - \hat{\beta}_j$

The function `fitted` calculates the inner level fitted values for an `lmer` object. Usage is

`fitted(`*lmobject*`)`

Correspondingly

`resid(`*lmobject*`)`

gives the inner level residuals.

Outer level fitted values and residuals need to be extracted manually: we multiply the design matrix for the fixed effect terms by the estimated fixed effects.

```
fitted.level0<-fm1@pp$X %*% fixef(fm1)
resid.level0<-effort-fitted.level0
```

Since the random errors ϵ_{ij} are estimated by the subject-wise residuals $\hat{\epsilon}_{ij}$, we use here the obvious plots:

- **Subject-level Residuals v Fitted Values** – looking for random scatter,
- **Population-level Residuals v Fitted Values** – looking for random scatter,
- **Normal QQ Plot of the Subject-wise Residuals** – looking for a straight line.

The plots may be made in R by

```
plot(fitted(fm1), residuals(fm1))
plot(fitted.level0,resid.level0)
qqnorm(resid(fm1))
abline(0,1.1003)
```

The plots are shown in Figures 4.2 , 4.3. and 4.4.

We can also plot residuals versus fitted values by stool type.

```
xyplot(resid(fm1)~fitted(fm1)|Type)
```

This is shown in Figure 4.5.

To assess the general fit of the model one should examine a plot of **Response v (Subject-wise) Fitted Values** – looking ideally for a straight line with unit slope.

The plot is obtained by

```
plot(fitted(fm1),ergoStool$effort)
abline(0,1)
```

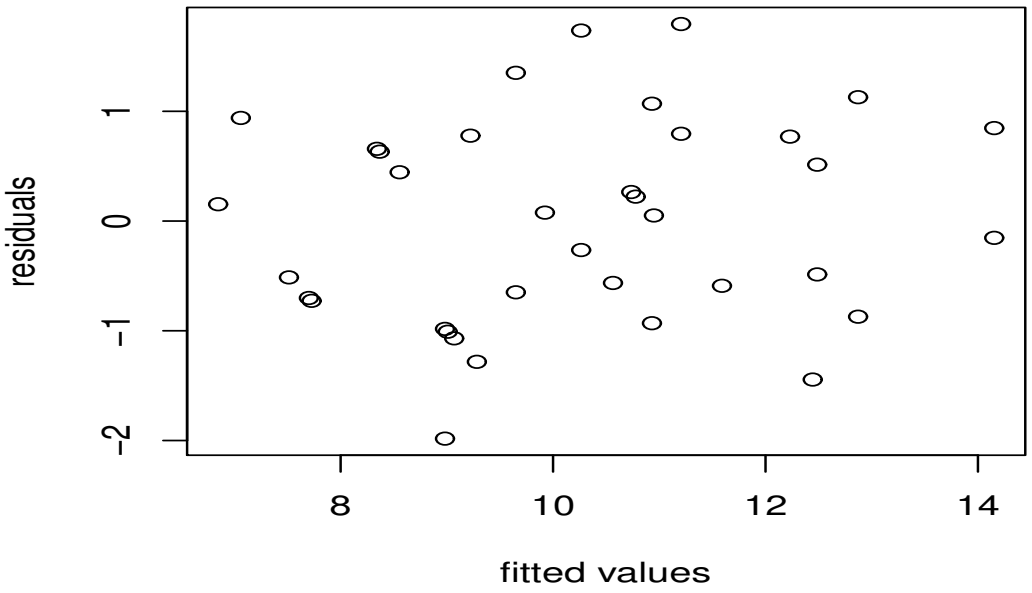

Figure 4.2: Within-group residuals versus fitted values for erg1.lme

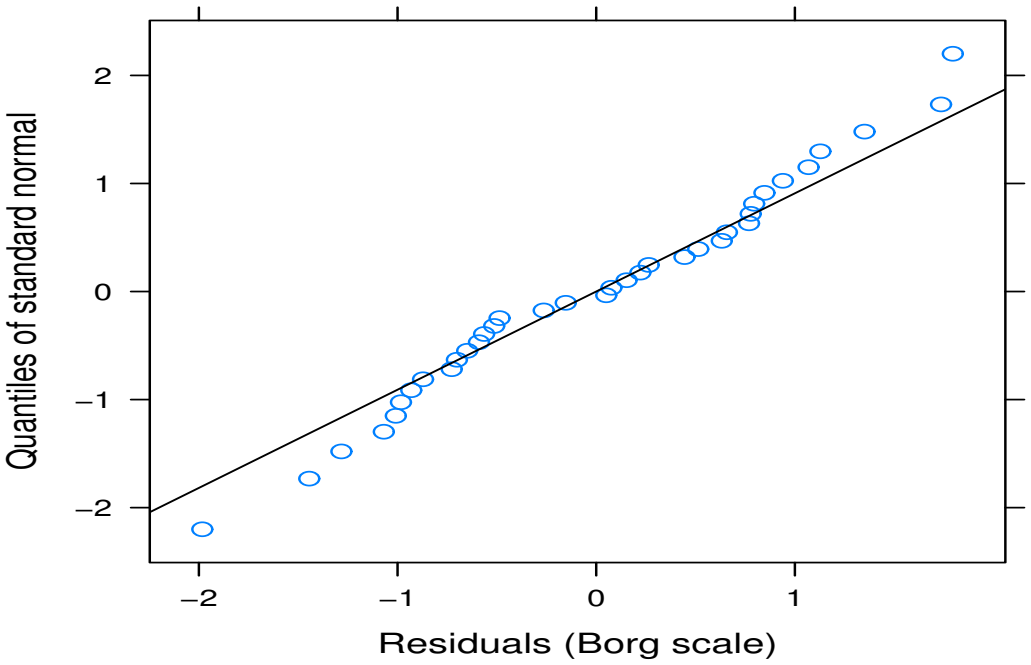

Figure 4.3: QQnorm plot of within-group residuals for erg1.lme

and can be seen in Figure 4.6.

4.1.1 Oxides example revisited

Recall the `oxides` dataset and the model

$$\begin{aligned} Y_{ijk} &= \beta + b_i + b_{ij} + \epsilon_{ijk}, \quad i = 1, ..., 8, \quad j, k = 1, 2, 3, \\ & \quad b_i \sim N(0, \sigma_1^2), \quad b_{ij} \sim N(0, \sigma_2^2), \quad \epsilon_{ijk} \sim N(0, \sigma^2) \end{aligned} \tag{4.3}$$

where Y_{ijk} represents the oxide thickness for the kth site on the jth wafer from the ith lot, β is the mean thickness (fixed effect), b_i is the random effect of the ith lot, b_{ij} is the random effect of the jth wafer from the ith lot and ϵ_{ijk} is the random error for the experimental units.

Note that we now have three distributional assumptions to check (two Normal distributions for the random effects and 1 for the experimental error):

- $b_i \sim N(0, \sigma_1^2)$; - assumption 1
- $b_{ij} \sim N(0, \sigma_2^2)$; - assumption 2
- $\epsilon_{ijk} \sim N(0, \sigma^2)$; - assumption 3

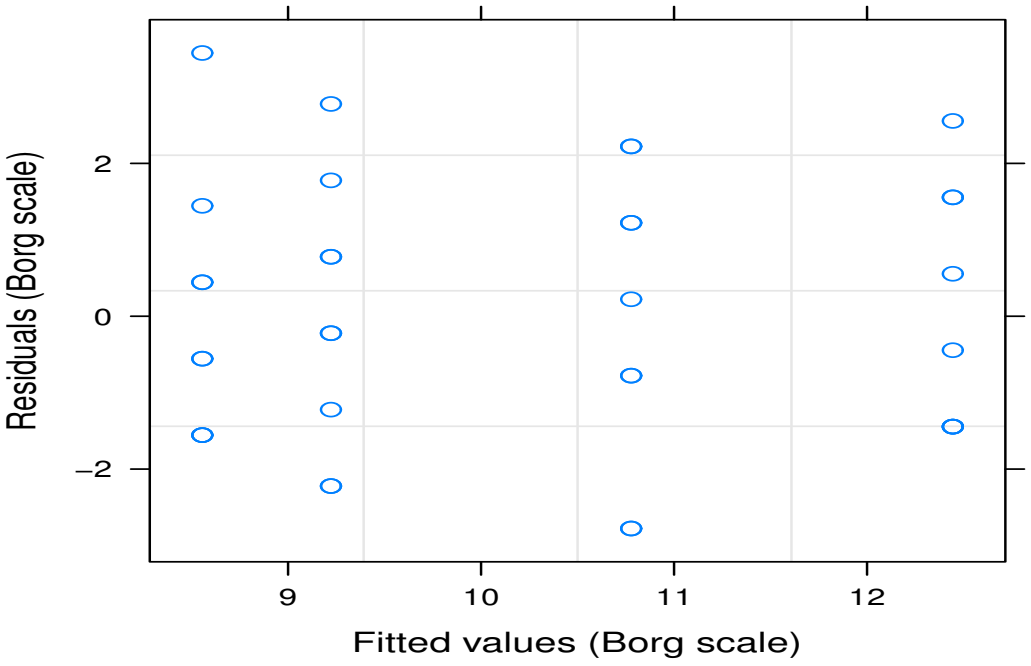

Figure 4.4: Population-level residuals versus fitted values for erg1.lme

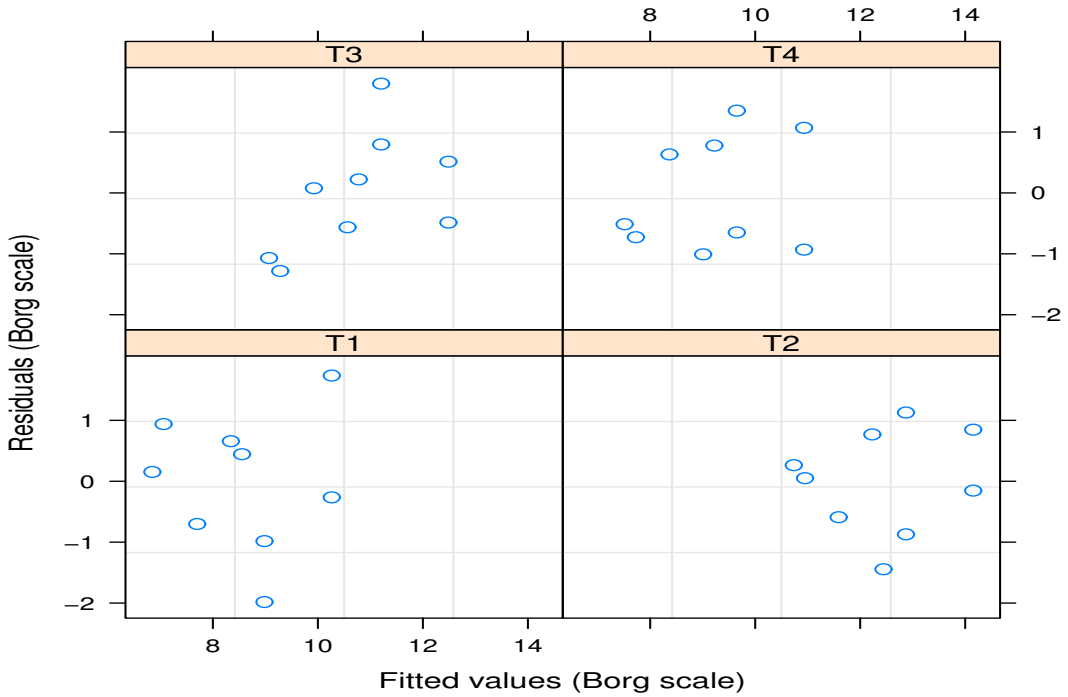

Figure 4.5: Within-group residuals versus fitted values by stool type for erg1.lme

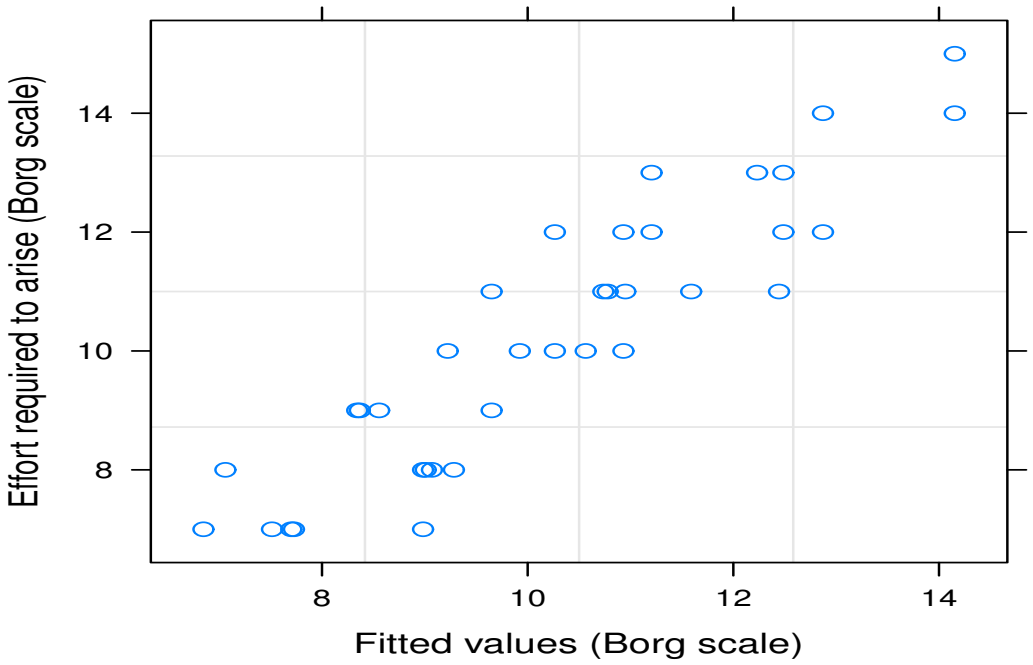

Figure 4.6: Adequacy of fixed effects structure for erg1.lme

Both sets of estimated random effects are obtained using the `ranef` command. We first examine the values of the random effects and their normality in 1 and 2 using the plots:

```
qqnorm(unlist(ranef(fm1)$‘Wafer:Lot‘),ylab="Wafer-within-lot-level random effects")
abline(0,5.9891)

qqnorm(unlist(ranef(fm1)$Lot),ylab="Lot-level random effects")
abline(0,11.3967)
```

Figure 4.7 is a cause for concern as it shows evidence of departure from normality although, as is often the case with random effects, there are a small number of observations.

To plot residuals against fitted values (at the wafer-within-lot level), we do

```
plot(fitted(fm1),residuals(fm1))
```

We can also produce separate residual plots for each wafer, using.

```
xyplot(residuals(fm1)~fitted(fm1) | Wafer)
```

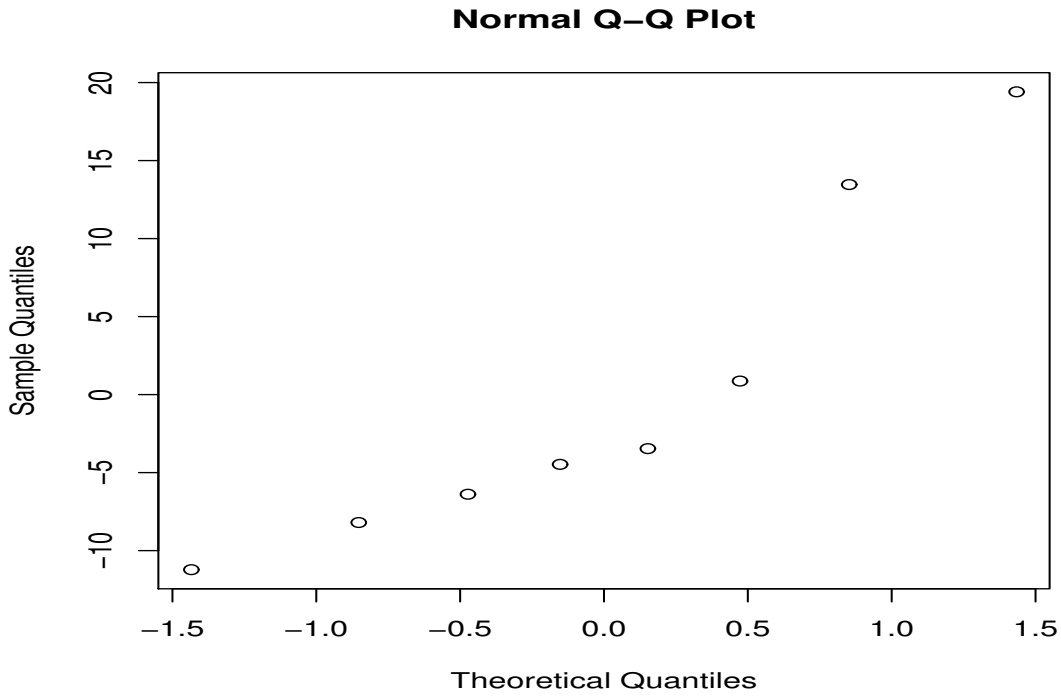

Figure 4.7: QQ plot of lot-level random effects for oxide.lme

Task 10. *Investigate the model assumptions for model* `fm1` *in* `Machines` *dataset, described in Section 3.2.*

4.2 Model Comparions: Comparing Fixed Effect Structures

Usually one compares models by assessing the difference in the quality of fit through a generalized likelihood ratio test (GLRT). Goodness of fit measures based on maximised log-likelihoods such as the AIC or BIC can also be compared for different models. However, if the REML method is used to fit the model, the restricted likelihood obtained is dependent on the parameterization used for the fixed effects, and so it cannot be used to compare fixed effects structures.

Three options are

- Using the standard χ^2 to the log of the likelihood ratio, approximation according to Wilks's theorem (as long as fitting has been done by ordinary maximum likelihood)

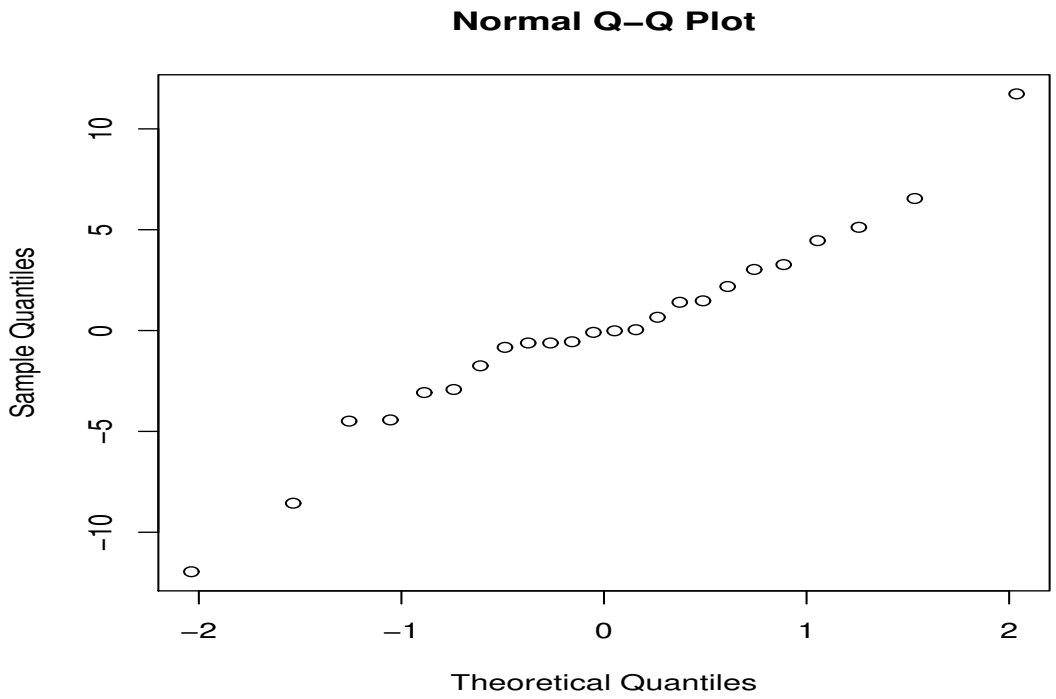

Figure 4.8: QQ plot of wafer-within-lot-level random effects for oxide.lme

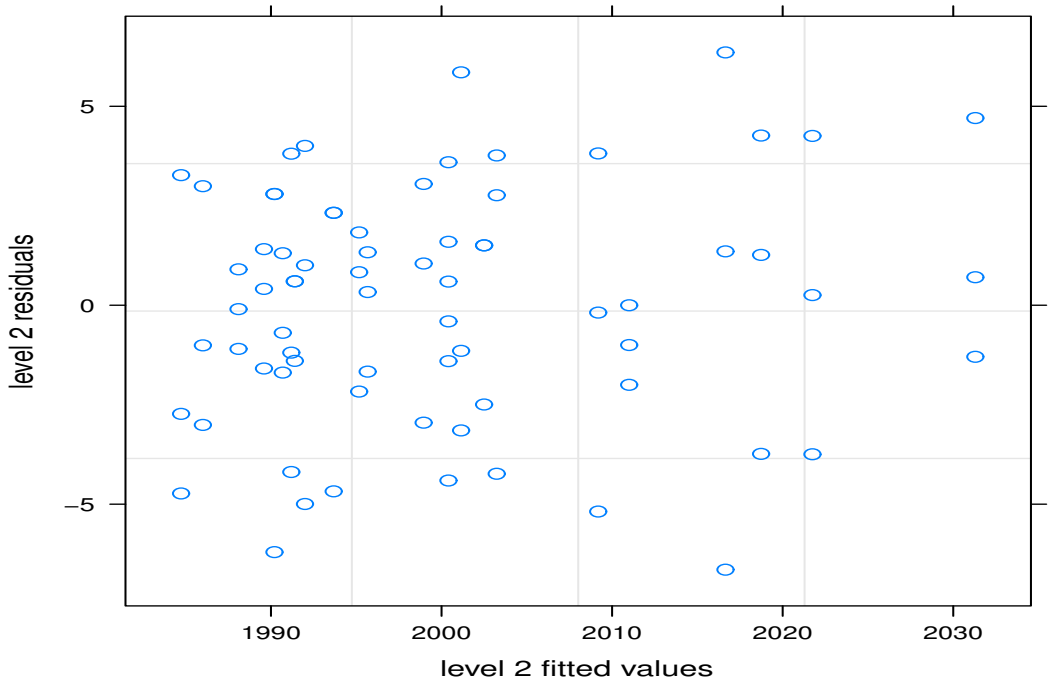

Figure 4.9: Level 2 residuals for oxide.lme

- Using an F-test, conditional on estimates of the parameters in the distribution of random effects;
- Using a parametric bootstrap hypothesis test, based on the GLRT.

Regarding option 1, Pinheiro & Bates (see their Section 2.4.2) declare that the resulting test (using the standard χ^2 asymptotic reference distribution) can sometimes be badly anti-conservative, that is, it can produce p-values that are too small. Wilks's theorem assumes that the estimated parameters are not on the boundary of the parameter space. For mixed effects models, often we find that once of the variance components is essentially zero, or that the models are not properly nested, thus violating the conditions of the theorem. Option 2 is derived from the classical approach and is implemented in some software packages. For balanced designs, we will get the same results as using an F-test in a fixed effects model. Otherwise, determining the appropriate degrees of freedom to use in the F-test is not straightforward, and we can't justify that the test-statistic will have an F-distribution in any case. The `lme4` package deliberately does *not* give a p-value! Here we consider option 3. This is computationally expensive, but has the advantage that we do

not have to assume a distribution for the test statistic. We first revise the GLRT, before introducing bootstrapping.

4.2.1 The generalized likelihood ratio test

We use the following notation. Suppose observations are independent and the ith observation is drawn from the density (or probability function, if discrete) $f_i(x, \psi)$ where ψ is a vector of unknown parameters. Then the likelihood and log-likelihood of the observations $y = (y_i)$ are

$$L(y, \psi) = \prod_i f_i(y_i, \psi) \quad \text{and} \quad l(y, \psi) = \log L(y, \psi) = \sum_i \log f_i(y_i, \psi) \tag{4.4}$$

respectively.

Now suppose that a simplification is proposed that entails imposing r restrictions, $s_i(\psi) = 0$ for $i = 1, \ldots, r$, on the elements of ψ, where r is less than the number of components in ψ. These restrictions must be functionally independent, which is another way of saying all r of them are needed. Suppose the maximum likelihood for the restricted situation is $\widehat{\psi}_s$, which is obtained by maximising l subject to the constraints. Then the generalised likelihood ratio test (GLRT) considers the test statistic

$$L := -2 \log \Lambda = -2 \left(l(y, \widehat{\psi}_s) - l(y, \widehat{\psi}) \right) \tag{4.5}$$

That is, twice the difference between the maximised value of the log-likelihood from the simpler model and that for the more complex one. If the simpler model is correct (i.e. H_0 is true), then we expect to see smaller values of $-2 \log \Lambda$. Whereas, if the simpler model is false, we expect this difference to be larger. This forms the basis of a test called the generalized likelihood ratio test (GLRT).

To summarise, to conduct a GLRT, we do the following.

1. Evaluate the maximised log-likelihood $l(y, \hat{\psi})$ under the alternative hypothesis.
2. Evaluate the maximised log-likelihood $l(y, \hat{\psi}_s)$ under the null hypothesis.
3. Evaluate the test statistic

$$L_{obs} = -2 \left(l(y, \hat{\psi}_s) - l(y, \hat{\psi}) \right). \tag{4.6}$$

and decide whether L_{obs} is larger than expected under H_0.

We have various options to decide if L_{obs} is extreme or not. According to Wilks's theorem, under H_0

$$-2\log\Lambda \sim \chi^2_r \qquad \text{approx.} \tag{4.7}$$

and so in step 3. we would compare L_{obs} with the χ^2_r distribution, where r is the number of constraints specified by the null hypothesis. So, if for example we find $L_{obs} > \chi^2_{r;0.95}$, we would reject H_0 at the 5% level. However, as discussed above, the assumptions of this theorem do not always apply.

In some cases we can derive the exact distribution of L, rather than using the χ^2 approximation.

Task 11. *(Optional) Consider using a GLRT for the hypothesis $H_0 : \mu = 0$ against a two-sided alternative, given data $Y_1, \ldots, Y_n \overset{\text{iid}}{\sim} N(\mu, \sigma^2)$, where σ^2 is known. Show that the GLRT is equivalent to the usual Z test based on $\bar{Y}/(\sigma/\sqrt{n})$,*

Example

(For supporting R code, see the R script file `MAS474-GLRT-bootstrapping.R`). We illustrate the GLRT on an ordinary linear model. We use the GLRT to test the hypothesis $H_0 : \beta = 0$ in the model

$$Y_i = \alpha + \beta x_i + \varepsilon_i,$$

for $i = 1, \ldots, n$ with $\varepsilon_i \sim N(0, \sigma^2)$. (In fact, the usual F-test/ t-test for this hypothesis *is* a GLRT, but one in which we can derive the exact distribution of L.)

The set of parameters ψ in the alternative hypothesis are α, β and σ^2, and the constraint applied by the null hypothesis is $\beta = 0$, so $r = 1$. Hence to apply the GLRT we do the following.

1. Find the maximum likelihood estimates of α, β, σ^2 in the full model.
$$Y_i = \alpha + \beta x_i + \varepsilon_i,$$
The m.l.e's of α and β are the same as the least squares estimates, but the m.l.e. of σ^2 is RSS/n. Evaluate the log-likelihood at these parameter estimates, and denote the maximised log likelihood by $l(y, \hat{\psi})$.

2. Find the maximum likelihood estimates of α, σ^2 in the reduced model
$$Y_i = \alpha + \varepsilon_i.$$
Evaluate the log-likelihood at these parameter estimates, and denote the maximised log likelihood by $l(y, \hat{\psi}_s)$.

3. Evaluate the test statistic

$$L = -2\left(l(y, \hat{\psi}_s) - l(y, \hat{\psi})\right), \tag{4.8}$$

and compare against the χ^2_1 distribution, rejecting H_0 if L is large in comparison.

In the R script file, you will see that the p-values for the GLRT and the usual F-test are different, because the χ^2_1 distribution is only an approximation for the distribution of L.

4.2.2 Bootstrapping

Bootstrapping will be covered in more depth in Computational Inference, but we give a short discussion here (specifically on *parametric* bootstrapping).

The idea is to use simulation to estimate the properties of a random variable, in situations where we cannot derive the distribution analytically. Consider again the model

$$Y_i = \alpha + \beta x_i + \varepsilon_i,$$

for $i = 1, \ldots, n$ with $\varepsilon_i \sim N(0, \sigma^2)$, the null hypothesis $H_0 : \beta_0$ and the corresponding GLRT statistic

$$L = -2\left(l(y, \hat{\psi}_s) - l(y, \hat{\psi})\right). \tag{4.9}$$

If L really does have a χ^2_1 distribution, then we can calculate probabilities $P(L \geq c)$ for any value of c, so that we can get a p-value for our hypothesis test (in R, use `1-pchisq(c,1)`). But the χ^2_1 distribution is only an approximation, and not necessarily a good one. How else might we determine $P(L \geq c)$? Suppose we can *simulate* random values of $L = -2\log\Lambda$ from its distribution under H_0. Given a sample $L_1, \ldots, L_N$, we can estimate $P(L \geq c)$ by the proportion of values in the sample greater than c. Write this proportion as $\frac{\sum_{i=1}^N I(L_i \geq c)}{N}$. By definition, $P(L_i \geq c) = P(L \geq c)$, so that

$$\begin{aligned}
\sum_{i=1}^{N} I(L_i \geq c) &\sim Binomial\{N, P(L \geq c)\} \\
\Rightarrow E\left\{\frac{\sum_{i=1}^N I(L_i \geq c)}{N}\right\} &= P(L \geq c) \\
\text{and } Var\left\{\frac{\sum_{i=1}^N I(L_i \geq c)}{N}\right\} &= \frac{P(L \geq c)(1 - P(L \geq c))}{N},
\end{aligned}$$

so that, for large N, the estimate of $P(L \geq c)$ (and hence the desired p-value) should be close to the true value.

Simulating the test statistic

We need to simulate values of L *under the condition that* H_0 *is true*. One way to do this is to fit the reduced model to the data, then simulate new values of $Y_1, \ldots, Y_n$ given the original values of $x_1, \ldots, x_n$, assuming the parameters are equal to their estimated values. The algorithm is therefore

1. Calculate the GLRT statistic for the observed data. Denote the value of the statistic by L_{obs}.
2. Fit the reduced model $Y_i = \alpha + \varepsilon_i$ to the data, obtaining parameter estimates $\hat{\alpha}$ and $\hat{\sigma}^2$
3. Simulate new values of $Y_1, \ldots, Y_n$, assuming $\alpha = \hat{\alpha}$ and $\sigma^2 = \hat{\sigma}^2$. (This sample is referred to as a *bootstrapped* data sample).
4. For the simulated data, calculate the GLRT statistic
$$L = -2\left(l(y, \hat{\psi}_s) - l(y, \hat{\psi})\right). \tag{4.10}$$
5. Repeat steps 3-4 N times, to get a sample of GLRT statistics $L_1, \ldots, L_N$.
6. Estimate the p-value by
$$\frac{\sum_{i=1}^{N} I(L_i \geq L_{obs})}{N}.$$

Does it matter that we have used *estimated* parameters when simulating the new data? No, because (after some effort!) it is possible to show that the distribution of L doesn't depend on the parameter choices in the reduced model (as long as $\sigma^2 > 0$), so we could actually have chosen any parameter values for step 3.

Bootstrapping for confidence intervals

We can also use bootstrapping to get approximate confidence intervals. Informally, we can think of constructing a confidence interval through understanding how far a parameter estimate is likely to lie from its true value. For example, if we estimate a parameter θ by some function of the data $\hat{\theta}$, and we know there is a 95% chance that $\hat{\theta}$ will lie within a distance of k from θ, then the random interval $(\hat{\theta} - k, \hat{\theta} + k)$ must have a 95% chance of containing θ. We can use simulation to learn approximately the distribution of an estimator around the true parameter.

Suppose, for the simple linear regression model, we wish to obtain 95% confidence intervals for α, β and σ^2. A parametric bootstrapping procedure is as follows.

1. Fit the full model $Y_i = \alpha + \beta x_i + \varepsilon_i$ to the data, obtaining parameter estimates $\hat{\alpha}$, $\hat{\beta}$ and $\hat{\sigma}^2$.
2. Simulate new values of $Y_1, \ldots, Y_n$, assuming $\alpha = \hat{\alpha}$, $\beta = \hat{\beta}$ and $\sigma^2 = \hat{\sigma}^2$.
3. For the simulated data, re-fit the model, and get new parameter estimates $\hat{\alpha}_i$, $\hat{\beta}_i$ and $\hat{\sigma}_i^2$
4. Repeat steps 2-3 N times.
5. Find the 2.5th and 97.5th sample percentiles within each set $\{\hat{\alpha}_1, \ldots, \hat{\alpha}_N\}$, $\{\hat{\beta}_1, \ldots, \hat{\beta}_N\}$, and $\{\hat{\sigma}_1^2, \ldots, \hat{\sigma}_N^2\}$, and report these as the confidence intervals.

For small samples, you may find this gives narrower intervals compared with those found using the usual method, as no 'adjustment' (in the sense of switching to a normal to a t-distribution) is made to allow for using an estimate of σ^2 in step 1. For larger n, where $t_{n-2;0.975} \simeq Z_{0.975}$, bootstrapping will give similar results to those derived from standard theory.

To repeat what we said at the start, these methods are *not* necessary for the analysis of standard linear models (assuming the model assumptions hold), but they *are* useful for more complex models where we can't derive the necessary results analytically.

Task 12. *Download and work through the R script* `MAS474-GLRT-bootstrapping.R`

4.2.3 Comparing fixed effects structures with a bootstrap hypothesis test: an example

We consider again the `ergostool` dataset. Suppose we wish to test the hypothesis $\beta_j = \beta$ for $j = 1, \ldots, 4$, so that the full model is

$$Y_{ij} = \beta_j + b_i + \epsilon_{ij}, \quad i = 1, \ldots, 9, \quad j = 1, \ldots, 4, \tag{4.11}$$

and the reduced model is

$$Y_{ij} = \beta + b_i + \epsilon_{ij}, \quad i = 1, \ldots, 9, \quad j = 1, \ldots, 4. \tag{4.12}$$

We use the GLRT test statistic, but use simulation to estimate its distribution under H_0. The procedure is as follows.

1. Fit the full model and obtain the log-likelihood $l(y, \hat{\psi})$.

```
> fm.full<-lmer(effort~Type + (1|Subject),ergoStool,REML=F)
> logLik(fm.full)
'log Lik.' -61.07222 (df=6)
```

2. Fit the reduced model and obtain the log-likelihood $l(y, \hat{\psi}_s)$.

```
> fm.reduced<-lmer(effort~1 + (1|Subject),ergoStool,REML=F)
> logLik(fm.reduced)
'log Lik.' -79.07502 (df=3)
```

3. Calculate the observed test statistic

$$L_{obs} = -2\left(l(y, \hat{\psi}_s) - l(y, \hat{\psi})\right).$$

```
> (obs.test.stat<- - 2*(logLik(fm.reduced)-logLik(fm.full)) )
[1] 36.0056
```

4. Simulate new data from the reduced model, fit both models, and re-calculate the test statistic. Repeat this process N times.

```
N<-1000
sample.test.stat<-rep(0,N)
for(i in 1:N){
new.y<-unlist(simulate(fm.reduced))
fm.reduced.new<-lmer(new.y~1 + (1|Subject) , REML=F)
fm.full.new<-lmer(new.y~Type + (1|Subject), REML=F)
sample.test.stat[i]<- -2*(logLik(fm.reduced.new)-logLik(fm.full.new))
}
```

5. Count the proportion of times the simulated test statistic is greater than the observed test statistic

```
> mean(sample.test.stat>=obs.test.stat)
[1] 0
```

In this case, as the observed test statistic was very large, *all* the simulated test statistics were smaller, suggesting a p-value smaller than 0.001. (We can increase N, but we shouldn't worry too much trying to find a precise estimate of a very small p-value).

4.2.4 Confidence intervals

Constructing confidence intervals is not entirely straightforward, as we again have the problem of determining distributions of estimators. The `lme4` command provides estimated standard errors for each fixed effect.

```
Fixed effects:
            Estimate Std. Error t value
(Intercept)   8.5556     0.5760  14.854
TypeT2        3.8889     0.5187   7.498
TypeT3        2.2222     0.5187   4.284
TypeT4        0.6667     0.5187   1.285
```

An approximate 95% interval is given by the estimate ± 2 estimated standard errors (assuming a normally distributed estimator). We can also use bootstrapping. We repeatedly simulate data from our chosen model, calculate new parameter estimates each time, and inspect the distribution of parameter estimates. To get confidence intervals for each `Type`, include a `-1` term in the fixed effect specification.

```
fm1<-lmer(effort~Type -1 + (1|Subject),ergoStool)
N<-1000
fixed.effects<-matrix(0,N,4)
variance.components<-matrix(0,N,2)

for(i in 1:N){
new.y<- unlist(simulate(fm1))
new.fm1 <- lmer(new.y~Type -1 + (1|Subject))
fixed.effects[i,] <- fixef(new.fm1)
vc <- VarCorr(new.fm1)
variance.components[i,]<-c(unlist( lapply(vc, diag) ),attr(vc,"sc")^2)
# Element [i,1] is random effect variance
# Element [i,2] is residual variance
}
```

Then to get confidence intervals

```
> apply(fixed.effects,2,quantile,probs=c(0.025,0.975))
          [,1]     [,2]      [,3]      [,4]
2.5%  7.476842 11.34321  9.707421  8.044497
97.5% 9.607266 13.63522 11.863876 10.334673

> apply(variance.components^0.5,2,quantile,probs=c(0.025,0.975))
           [,1]      [,2]
```

```
2.5%  0.5182556 0.8036279
97.5% 2.0350936 1.4207169
```

The approximate 95% confidence interval for σ_b is (0.5,2.0) and the approximate confidence interval for σ is (0.8, 1.4). It is important to appreciate that these bootstrap intervals are only approximate, so we should not interpret the values too precisely, but they are indicative of the precision of the various parameter estimates.

4.2.5 Comparing Random Effects Structures

We have seen models where different random effects structures are possible, for example in the `Machines` dataset. In general, one would retain all grouping variables in a model (even if they appeared to have little effect) on the grounds that they were necessary to account properly for the structure of the experiment, but one might certainly wish to consider whether interactions with them are necessary.

The method implemented in `lme4` is the GLRT. If we parametrize so that Model 2 is the more general model and we write L_i for the likelihood and k_i for the number of parameters in model i, then asymptotically (as sample sizes grow large) under the hypothesis that the data were in fact generated from the more restrictive model, Model 1, we have

$$2[\log(L_2) - \log(L_1)] \sim \chi^2_{k_2-k_1}. \tag{4.13}$$

The difference of log-likelihoods tends to be larger when the hypothesis is false, and so a significance test is obtained by comparing values of the statistic with quantiles of the χ^2 distribution.

Pinheiro & Bates (see their Section 2.4.1 for details), report that this asymptotic result may produce **conservative** tests (that is, p-values larger than they should be, so tests that are overprotective of H_0) because the change from the more general to more specific model involves setting the variance of some of the random effects components to zero, which is on the boundary of the parameter region. They investigate, by simulation, some suggested modifications, but find that they are not always beneficial. They conclude that the appropriate means of comparing random effects structures is to use the standard generalized likelihood ratio test, but be aware in interpreting results that it might be conservative.

The GLRT is implemented in R using the `anova` command with the two models as arguments (the `anova` command will refit the

models using ordinary ML, if REML has been used). The output shows the value of the GLRT statistic and gives the p-value from the $\chi^2_{k_2-k_1}$ distribution.

If we do this for the `Machines` data set we get

```
> fm.full<-lmer(score~Machine+(1|Worker/Machine),data=Machines)
> fm.reduced<-lmer(score~Machine+(1|Worker),data=Machines)
> anova(fm.reduced,fm.full)
Data: Machines
Models:
fm.reduced: score ~ Machine + (1 | Worker)
fm.full: score ~ Machine + (1 | Worker/Machine)
           Df    AIC   BIC  logLik  Chisq Chi Df Pr(>Chisq)
fm.reduced  5 303.75 313.7 -146.88
fm.full     6 237.47 249.4 -112.73 68.289      1  < 2.2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

So overwhelming evidence against the null that the coefficients of the random interaction terms are zero.

Task 13. *Consider again the* `Orthodont` *data discussed in Section 3.3. Perform a GLRT to compare the model with random intercepts with that including both random intercepts and random slopes.*

Revision Checklist

We have now finished the material on mixed effects models. To help with your revision, here is a short list of concepts/methods that you should know and understand.

1. The meaning of a fixed effect and a random effect in a linear model. When to use each.
2. How to write down a mixed effects model, for example, with one factor modelled as a fixed effect, and one factor modelled as a random effect.
3. How to work out covariances between observations in a mixed effects model.
4. Fitting a mixed effects model in R, and interpreting the output.
5. The concept of REML, and how it is used to estimate parameters in a mixed effects model.
6. How random effects are predicted using BLUP.
7. Checking model assumptions in a mixed effects model.
8. How to use bootstrapping to compare two nested fixed effects models, and to obtain approximate confidence intervals.
9. How to use the GLRT to compare different random effects structures.

Chapter 5

Missing data

5.1 Introduction

Missing data are observations we intended to make but didn't. For example, this can be due to

- Non-response in a survey
- Censoring, e.g., your instrument doesn't record beyond a certain range
- Corrupted/lost data
- Attrition in longitudinal studies, etc

Missing data present a problem for statistical analyses as if we simply ignore the problem, for example, by analysing complete cases only, we may reach biased conclusions, or conclusions that lack power.

In this part of the module, we will examine methods for dealing with missing data that can be used in a range of situations. We will begin by introducing some types of missingness, and then describe some simple but flawed approaches for dealing with it. We will then look at likelihood based approaches, focussing on the use of the EM algorithm. Finally, we will look at a more widely applicable approach called multiple imputation, which works by proposing values for the missing data ('imputing' it), conducting the analysis on the imputed data, repeating this multiple times to generate multiple estimates, and then combining these estimates in an appropriate way.

The notes are based largely on the text book by Little and Rubin

- *Statistical analysis with missing data*, Second Edition, R. J. A. Little and D. B. Rubin, Wiley, 2002.

but in Chapter 7, we also make use of the paper by van Buuren and Groothuis-Oudshoorn

- S. van Buuren and K. Groothuis-Oudshoorn, *mice: Multivariate Imputation by Chained Equations in R*, , Journal of Statistical Software, 45(3), 2011.

5.2 Mechanisms of missingness

It is important to understand why data is missing, and then to adjust the analysis to take this into account. We will suppose that $Y = (y_{ij})$ is a $n \times k$ matrix containing the data. Typically, for example, it will consist of k pieces of information on each of n different subjects. We define the missing data indicator matrix $M = (m_{ij})$ to consist of values

$$m_{ij} = \begin{cases} 1 & \text{if } y_{ij} \text{ is missing} \\ 0 & \text{otherwise} \end{cases}$$

so that M defines the pattern of missing data. Most statistical computing languages have special characters to denote missingness. In R, for example, `NA` is used to denote missingness.

We distinguish between three different mechanisms that lead to missing data. The difference between these depends on how the variables that are missing are related to the underlying values of the variables in the dataset. This relationship is characterized by the conditional distribution of M given Y,

$$f(M \mid Y, \phi),$$

say, where ϕ is any unknown parameters. Let Y_{obs} and Y_{mis} denote the observed and missing parts of Y respectively.

- If missingness does not depend on the value of the data, i.e., if

  $$f(M \mid Y, \phi) = f(M \mid \phi) \text{ for all } Y, \phi$$

  then the missing data mechanism is called **Missing Completely At Random (MCAR)**. This is the ideal situation.

- If missingness depends only on the observed components of Y, denoted Y_{obs}, so that

  $$f(M \mid Y, \phi) = f(M \mid Y_{obs}, \phi) \text{ for all } Y_{mis}, \phi$$

  then we call the missing-data mechanism **Missing At Random (MAR)**. In this case, the probability of being

missing can depend on the value of the other observed (not missing) covariates. For reasons which will become clear later, this type of missing data mechanism is often called **ignorable**.

- Finally, if the distribution of M depends on the missing values in Y, then we say the mechanism is **Not Missing At Random (NMAR)**. This case is hardest to deal with, and is non-ignorable, as it can lead to serious biases if not analysed carefully.

Note that MCAR is also MAR.

Usually it is impossible to tell from the data alone whether the missing observations are NMAR or MAR[1]. Thus we have to make assumptions (after thinking hard) about the missing mechanism.

Examples

1. Suppose we are conducting a survey. Answers would be MCAR if a randomly selected subset of respondents didn't answer Q10. If on the other hand, the probability of answering Q10 depends on answers for Q1-9, then the missing-data mechanism is MAR. Finally, if the probability the answer to Q10 is missing depends on the answer to Q10, for example, because it is embarrassing, then the mechanism is NMAR. For example, it has been observed that people's propensity to answer the question, *What is your annual salary?*, depends upon the answer, with high-earners less likely to answer the question than lower-earners.

2. Suppose $Y_i \sim N(0, 1)$. Then if

$$\mathbb{P}(M_i = 1 \mid y_i, \phi) = \frac{1}{2} \text{ for all } y_i$$

then the data are MCAR. In this case we can analyse the data using the non-missing values (complete case analysis - see below) without incurring any bias. For example, we will still find that $\mathbb{E}(\bar{y}_{obs}) = 0$. However, if the data are censored, for example with only negative values recorded, i.e.,

$$\mathbb{P}(M_i = 1 \mid y_i, \phi) = \begin{cases} 1 & \text{if } y_i \geq 0 \\ 0 & \text{otherwise,} \end{cases}$$

[1] We can often spot the difference between MCAR and MAR.

then the mechanism is clearly NMAR. If we now analyse the complete data only, ignoring the censoring mechanism, we will end up with biased estimators. It can be shown, for example, that

$$\mathbb{E}(\bar{y}_{obs}) = -\frac{1}{2(2\pi)^{0.5}}$$

in this case.

3. Attrition in longitudinal studies: In a longitudinal study we may often have attrition so that subjects drop out after some period, never to return. In this case, we observe a monotonic pattern of missingness, e.g., $y_{i1}, \ldots, y_{ij-1}$ all present but $y_{ij}, \ldots, y_{iK}$ missing for some j. In this case, we can record missingness with a single indicator M_i, which takes a value equal to the index of when subject i was first missing (so $M_i = j$ in the example here). In this case, the missing data mechanism is MCAR if

$$\mathbb{P}(M_i = j \mid y_{i1}, \ldots, y_{iK}, \phi) = \phi \; \forall \; y_{ij}$$

which is a strong assumption often violated. Data are MAR if dropout depends on values recorded prior to dropout, but not after, e.g., if

$$\mathbb{P}(M_i = j \mid y_{i1}, \ldots, y_{iK}, \phi) = \mathbb{P}(M_i = j \mid y_{i1}, \ldots, y_{ij-1}, \phi) \; \forall \; y_{ij}, \ldots, y_{iK}$$

For example, in a drug study, y_{ij} may be patient i's blood pressure at measurement point j. If the study protocol says that i must leave the study if their blood pressure goes above a certain value, then this would be a MAR mechanism, as missingness depends on observed past data, not on the future unobserved data.

5.3 Naive methods

For purposes of illustration, consider the data in Table 5.1. We will now consider several naive approaches for dealing with this missing data.

5.3.1 Complete-case (CC) analysis

The simplest approach to missing data is to simply analyse only those cases where all variables are present, discarding subjects who have missing data, i.e., remove i if $M_{ij} = 1$ for any j. This is known as *complete-case* analysis or *listwise deletion*.

Person	y_1	y_2
1	-6.6	-1.6
2	0.6	-2.0
3	0.2	0.6
4	NA	0.6
5	-3.3	NA

Table 5.1: Data on 5 subjects with two covariates, with two missing data values.

This can be a satisfactory approach in some situations where only a small fraction of the data is missing. For the toy data in Table 5.1, we would simply remove rows 4 and 5 from the data set.

Problems with this approach are that

- It is statistically inefficient (i.e., it doesn't use all of the available information)
- It can lead to biases in NMAR and MAR cases, leading to invalid inferences
- In situations where a large number of covariates each have some missing values, we may find we have only a few complete cases to work with.

For the data in Table 5.1, a complete case analysis would estimate the mean of y_1 to be $(-6.6 + 0.6 + 0.2)/3 = -1.933$, the mean of y_2 to be -1, and the covariance between y_1 and y_2 to be 0.325.

5.3.2 Available-case analysis

In situations where we record multiple pieces of information on each subject, complete case analysis can be wasteful for univariate analyses if we remove a subject from the analysis just because one of the covariates is missing. For example, in the above we removed row 4 and 5 when estimating the mean of y_1, even though $y_{1,5}$ was observed and provides useful information.

Available-case analysis includes all cases where the variable of interest is present. The advantage of this is that it uses all the available data. The disadvantage is that the sample base changes from variable to variable according to the pattern of missing data. This approach can also lead to problems when estimating

variance-covariance matrices, as it can lead to estimate that are not semi-positive definite, i.e., not valid covariance matrices.

For the data in Table 5.1, an available case analysis estimates the mean of y_1 to to be $(-6.6 + 0.6 + 0.2 - 3.3)/4 = -2.275$, the mean of y_2 to be -0.6, and $\mathbb{C}\text{or}(y_1, y_2) = 0.325$, which is the same as before as we have to remove rows 4 and 5 for the correlation estimate.

5.4 Single imputation methods

> "*The idea of imputation is both seductive and dangerous.*" Little and Rubin 2002

Complete case and available case analyses leave missing cases out of the analysis. An alternative approach is to fill-in, or **impute** missing values in the dataset, and then analyse the data as if they were complete. This approach is attractive because it allows us to sidestep all the difficulties that come from handling missing data. It is dangerous because it can, in some cases, lead to substantial biases in statistical analyses. Many types of imputation method are commonly used, some of which are better than others. We will now review some of these approaches.

5.4.1 Mean imputation

A simple approach is to replace missing data with the mean of the observed data for that variable (or some other deterministic quantity). So in the example, we'd substitute the value -2.275 into y_{41} and -0.6 into y_{52}, as this is the mean of the observed values of y_2. In longitudinal studies, we often use a slightly different imputation approach and simply carry forward the last observed value for a particular individual. For example, in a clinical trial looking at blood pressure, if a subject drops out in week 5, we would simply use their final blood pressure measurement and assume this stayed constant throughout the rest of the trial.

Once we have replaced the missing value, we then analyse the data as we would if there were no missing values. The problem with this approach is that it does not lead to proper measures of variance or uncertainty, and so we tend to be overly confident in our predictions.

If we apply this approach to these data, we would estimate the mean of y_1 to be -2.275, the mean of y_2 to be -0.6, and the covariance between y_1 and y_2 to be 0.232. Notice how the mean

estimates haven't changed, but that the correlation estimate has. Mean imputation will always under-estimate variances and covariances in this way (**Task:** Why?).

This method is thus not recommended in general (although it is often used!).

5.4.2 Regression imputation

Another approach to imputation is to build a regression model to predict the missing variables from the observed variables. For example, in the toy example, we would build a model of the form $y_2 = a + by_1 + \epsilon$, finding values $\hat{a} = -0.783$ and $\hat{b} = 0.112$ (estimated from the complete cases only). We would then substitute y_{52} with $\mathbb{E}(y_2|y_1 = -3.3, \hat{\beta}) = -0.783 + 0.112 \times (-3.3) = -1.15$. To impute the missing value of y_1 we would fit the model $y_1 = a + by_2 + \epsilon$, which suggests filling in the missing value with -0.43. If we then analyse the complete data we find the mean of y_1 to be -2.05, the mean of y_2 to be -0.566, and $\mathbb{C}\text{or}(y_1, y_2) = 0.307$. There are several problems with this approach.

1. It only uses the mean of the regression model prediction, that is, we are ignoring the random error part. This approach is consequently sometimes called *conditional mean imputation*.

2. It assumes the estimated parameter values are known without error, whereas in practice, we are highly uncertain about their values.

3. It relies on complete cases only to estimate the regression coefficients, and in some cases this can mean using only a tiny fraction of the data.

Even so, this approach can work well in a much wider range of settings than mean imputation, but the variability of imputed values will be too small, so the estimated precision of coefficients will be wrong and inferences misleading.

5.4.3 Implicit imputation methods

Some approaches are not explicitly stated, but instead are implicitly defined by an algorithm. For example

- **Hot deck imputation**, in which missing values are replaced by values from similar subjects. For example, we

could use a nearest neighbour approach, in which if y_{ij} is missing, we find the subject k in the data for which $y_{kj'}$ most closely matches $y_{ij'}$ for $j' \neq j$ according to some metric.

Alternatively we might use a random sampling scheme, and replace the missing value y_{ij} by selecting at random a value y_{kj}, perhaps limiting the sampling space to subjects k that fall in a similar subset to case i. For example, if we are doing a survey on school children, and subject i is a girl in year 2, we might impute missing values about this child by randomly sampling the value with replacement from the girls in year 2 that did respond.

- **Cold deck imputation** replaces missing values with a constant value from an external source.

In general, although these approaches can sometimes result in satisfactory answers in certain cases, they are not generally recommended.

5.4.4 Stochastic imputation

The mean imputation methods discussed above (including conditional mean/regression imputation) tend to lead to underestimates of uncertainties, such as variances and covariances, and p-values and confidence intervals will be too small. This is particularly worrying when tails of distributions or standard errors are the primary object of interest. For example, if we are studying poverty, which concerns the lower tail of the income distribution, we find that imputing conditional means for missing incomes tends to underestimate the percentage of people living in poverty. This is because best prediction imputations systematically underestimate variability.

An alternative is to use stochastic (i.e. random) imputed values. For example, in the regression imputation model considered in Section 5.4.2, instead of filling in the missing value of y_{52} with $\mathbb{E}(y_2|y_1 = -3.3, \hat{\beta})$, the conditional mean prediction, we could instead use a random draw and substitute the value

$$-0.783 + 0.112 \times (-3.3) + e \text{ where } e \sim N(0, s^2)$$

and where $s = 1.87$ is the residual standard error. We would then analyse the data as if it were complete.

A very simple stochastic imputation approach is to randomly select one of the observed values, and to impute this for the missing

value. This will tend to reduce the covariance between different covariates, and so isn't recommended except in situations where you have a very small amount of missing data.

While conditional stochastic imputation is better than conditional mean imputation, it still doesn't account for imputation uncertainty. If we were to repeat the stochastic imputation above, we would get a different value, and thus different statistical estimates of whatever quantity we are interested in. In Chapter 7, we will look at an approach called **multiple imputation**, which generates several complete datasets, all with stochastic imputed values, and then analyses each of these separately and combines the estimates.

5.5 Missing data in regression problems

Consider the case where we have dependent variable y and covariates X. If X is complete, and values of y are missing at random (i.e., missingness depends on the covariates X only), then the incomplete cases contribute no information to the regression of y on X. If regression is the primary aim, then we may as well leave these observations out of the inference.

If missing values occur in elements of X, but in cases where y is observed, then it can be beneficial to try to account for this missingness in some way as there may be useful information in y that we wish to exploit. A common scenario is where X consists of k variables, $x_1, \ldots, x_k$ say, with a non-regular pattern of missingness, e.g., x_1 missing in some cases, x_2 and x_5 missing in others, etc. In this case, we can build several different regression models to predict the missing values. For example, we can predict the missing x_1 value using the observed values of $x_2, \ldots, x_k$ and y. Although it can appear circular to impute missing values of the covariates using y, only then to use these imputed values to build a model to predict y, it has been shown that this approach gives consistent estimates of the regression coefficients. We can also choose to impute x_1 on the basis of $x_2, \ldots, x_k$ alone (i.e. not using the observed value of y), and this also gives consistent estimates, but this approach can be significantly less efficient as it ignores the information available in y.

Note that if the missing-data mechanism is ignorable, the incomplete observations do not contain information about the regression parameters, i.e., if x_{i1} is missing, then case i can tell us nothing about β_1 (assuming our model is $y = \beta_0 + \beta_1 x_1 + \ldots \beta_k x_k + \epsilon$). So why bother imputing a value for x_{i1} or using a sophisticated

likelihood approach? It is because the information in case i does (potentially) tell us something useful about the other regression coefficients. We don't know how to fit the model with only partial observations for some cases (although that is the topic of the next chapter), so we would end up leaving all of case i out of the analysis if we don't impute its value.

Chapter 6

Missing data: Likelihood based methods

In the previous chapter we looked at methods that impute the missing values. We will return to these methods in Chapter 7. But now we will consider a likelihood based approach. We will proceed by specifying a multivariate statistical model of the data, and then use a maximum likelihood approach to estimate quantities of interest. The advantages of this approach are that it avoids the use of ad hoc imputation methods (and the problems this causes), and the result of the analysis takes into account the fact that data was missing, leading to consistent estimators of all quantities, including variances, p-values etc, which can be poorly estimated in imputation approaches. A further advantage is that this approach is very clear, in the sense that all our modelling assumptions are explicitly stated, and can be examined and changed. In contrast, imputation approaches usually use a variety of methods, which when combined do not necessarily result in a coherent model, and as a result, it can be difficult to understand the consequences of these methods in the final analysis.

The disadvantage of the likelihood based methods we are about to introduce are that they require us to carefully specify a statistical model for the data, when sometimes, we don't wish to have to think that hard about the problem. A further disadvantage is that the approach can be very involved, and sometimes requires complex algebraic calculations to be performed.

Before we begin this section, let's briefly recap marginal, joint and conditional probability density functions (pdfs). Suppose X and Y are two random variables with joint pdf $f(x, y)$. Then

the marginal distribution of X has pdf

$$f(x) = \int f(x, y)\mathrm{d}y.$$

In some cases we won't need to do the integral. For example, if

$$\begin{aligned} X \mid Y &\sim N(Y, \sigma^2) \\ Y &\sim N(0, \tau^2) \end{aligned}$$

then we can see that the marginal distribution of X is $N(0, \sigma^2 + \tau^2)$. In other cases we may need to calculate the integral. The conditional pdf of X given $Y = y$ is

$$f(x|y) = \frac{f(x, y)}{f(y)} = \frac{f(x, y)}{\int f(x, y)\mathrm{d}y}$$

and is proportional to $f(x, y)$ if we consider it to be just a function of x. We can thus see that

$$f(x, y) = f(x|y)f(y) = f(y|x)f(x).$$

6.1 Ignorable missing-data mechanisms

When do we need to model the missing-data mechanism? Modelling is hard, as it is not always obvious what an appropriate model is, and so if we can avoid modelling the missing data mechanism, $f(M|y, \phi)$, then we would like to do so. What we will show is that if the missing data mechanism is MAR (or MCAR) then we do not need to specify $f(M|y, \phi)$, but that if it is NMAR, then we do need to do so. MAR is a considerably weaker assumption than MCAR, and it is possible to test between these two situations, for example, by building a model to predict missingness and seeing whether there is a significant dependence upon the other covariates. It is impossible to test whether a missing data mechanism is MAR or NMAR, because the information we would need to make that judgement, namely Y_{mis}, is missing!

Let's begin by considering what happens if we ignore the missing-data mechanism. Consider the marginal probability density of the observed data:

$$\begin{aligned} f(Y_{obs} \mid \theta) &= \int f(Y_{obs}, Y_{mis} \mid \theta)\mathrm{d}Y_{mis} \\ &=: L_{ign}(\theta \mid Y_{obs}) \end{aligned}$$

which is the likelihood function for θ based on the observed data only. When will maximizing L_{ign} give us a valid estimator of θ?

There may, for example, be useful information in M that we can use to better learn θ. Or to look at it another way, ignoring the information contained in M may lead us to biased estimates and incorrect inferences.

Now consider the full model where we include a model of the missing-data mechanism. The joint distribution of Y and M is

$$f(Y, M \mid \theta, \psi) = f(Y \mid \theta) f(M \mid Y, \psi)$$

where ψ is a parameter for the missing-data mechanism, which may not be needed if this is known. Given the observed data (Y_{obs}, M), the full likelihood function is

$$\begin{aligned} L_{full}(\theta, \psi \mid Y_{obs}, M) &= f(Y_{obs}, M \mid \theta, \psi) \\ &= \int f(Y_{obs}, Y_{mis} \mid \theta) f(M \mid Y_{obs}, Y_{mis}, \psi) \mathrm{d}Y_{mis}. \end{aligned}$$

We can see that if the distribution of M doesn't depend on Y_{mis}, i.e., if it is MCAR or MAR, then

$$\begin{aligned} L_{full}(\theta, \psi \mid Y_{obs}, M) &= f(M \mid Y_{obs}, \psi) \int f(Y_{obs}, Y_{mis} \mid \theta) \mathrm{d}Y_{mis} \\ &= f(M \mid Y_{obs}, \psi) f(Y_{obs} \mid \theta) \\ &= f(M \mid Y_{obs}, \psi) L_{ign}(\theta \mid Y_{obs}) \end{aligned}$$

so that inference about θ based on L_{full} will be the same as inference based on L_{ign} as $L_{full} \propto L_{ign}$ (and so maximizing $L_{ign}(\theta)$ with respect to θ is equivalent to maximizing L_{full})[1]. For this reason **MAR** and **ignorable** are often used interchangeably.

6.1.1 Example: Censored exponential distribution

Suppose $y_i \sim Exp(1/\theta)$ for $i = 1, \ldots, n$, and that $y_1, \ldots, y_r$ are observed, but $y_{r+1}, \ldots, y_n$ are missing. We can show that the likelihood ignoring the missing-data mechanism is

$$L_{ign}(\theta) = f(y_1, \ldots, y_r \mid \theta) = \theta^{-r} \exp\left(-\frac{1}{\theta} \sum_{i=1}^{r} y_i\right).$$

We will consider two missing data mechanisms. Firstly, suppose each y_i is missing completely at random with probability ψ, so that

$$L_{full}(\theta, \psi \mid Y_{obs}, M) = \binom{n}{n-r} \psi^{n-r} (1-\psi)^r \theta^{-r} \exp\left(-\frac{1}{\theta} \sum_{i=1}^{r} y_i\right)$$

[1] Note we also require θ and ψ to be distinct here, which they are in most cases.

Then we can see that whether we maximize L_{full} or L_{ign} we find the same estimator for θ, namely

$$\hat{\theta} = \frac{\sum_{i=1}^{r} y_i}{r}.$$

This is expected as the missing-data mechanism is MAR and thus ignorable.

Now suppose instead we are told that missingness is due to observations being censored at c, so that they are not recorded if $Y_i \geq c$. It is now obvious that the pattern of missingness contains information about the parameters that should be taken into account. If we were to use the likelihood that ignores the missing-data mechanism, L_{ign}, then $\hat{\theta} = \frac{\sum_{i=1}^{r} y_i}{r}$ will still be our maximum likelihood estimator of θ. It is easy to see that this will be biased to be too small. Because the mechanism is NMAR in this case, we must consider the full model for (Y, M) and use the full likelihood.

The probability for the missing mechanism is the degenerate model

$$\mathbb{P}(M_i = 1 \mid y_i) = \begin{cases} 1 & \text{if } y_i \geq c \\ 0 & \text{if } y_i < c \end{cases} \tag{6.1}$$

$$\mathbb{P}(M_i = 0 \mid y_i) = \begin{cases} 0 & \text{if } y_i \geq c \\ 1 & \text{if } y_i < c. \end{cases} \tag{6.2}$$

But of course, if $M_i = 1$ we won't have observed y_i, and so we can't use these equations directly. We need to calculate

$$\begin{aligned} \mathbb{P}(M_i = 1 \mid \theta) &= \int_0^\infty \mathbb{P}(M_i = 1 \mid y_i) f(y_i \mid \theta) \mathrm{d}y_i \\ &= \int_0^\infty \mathbb{I}_{y_i \geq c} \frac{1}{\theta} e^{-\frac{y_i}{\theta}} \mathrm{d}y_i \\ &= \int_c^\infty \frac{1}{\theta} e^{-\frac{y_i}{\theta}} \mathrm{d}y_i. \\ &= \mathrm{e}^{-\frac{c}{\theta}}. \end{aligned}$$

Then

$$\begin{aligned} f(Y_{obs}, M \mid \theta) &= f(Y_{obs} \mid \theta) f(M \mid Y_{obs}, \theta) \\ &= \prod_{i=1}^{r} \theta^{-1} \exp(-\frac{y_i}{\theta}) \prod_{i=1}^{r} \mathbb{P}(M_i = 0 \mid Y_{obs}) \prod_{i=r+1}^{n} \mathbb{P}(M_i = 1 \mid \theta) \\ &\text{as } \mathbb{P}(M_i = 1 \mid Y_{obs}, \theta) = \mathbb{P}(M_i = 1 \mid \theta) \text{ and } \mathbb{P}(M_i = 0 \mid Y_{obs}, \theta) = \mathbb{P}(M_i = 0 \mid Y_{obs}) \\ &= \theta^{-r} \exp\left(-\frac{1}{\theta} \sum_{i=1}^{r} y_i\right) \times 1 \times \prod_{i=r+1}^{n} \mathbb{P}(y_i \geq c \mid \theta) \\ &= \theta^{-r} \exp\left(-\frac{1}{\theta} \sum_{i=1}^{r} y_i\right) \exp\left(-\frac{1}{\theta} c(n-r)\right). \end{aligned}$$

When we maximise this it gives us the correct estimator

$$\hat{\theta} = \frac{\sum_{i=1}^{r} y_i + (n-r)c}{r}, \tag{6.3}$$

which is inflated compared to the incorrect maximum likelihood estimate based on L_{ign}. See the R code on MOLE for an illustration.

This is the only example of a NMAR/non-ignorable missing data mechanism that we are going to consider in this module. In general, NMAR data are hard to work with as they require us to have a good understanding of the missingness mechanism so that we can build a model of it. See Chapter 15 in Little and Rubin for more details.

6.1.2 Estimation

Whether we have an ignorable missing-data mechanism so that we only need consider

$$f(Y_{obs} \mid \theta)$$

or a non-ignorable mechanism so that we need to instead consider

$$f(M \mid Y_{obs}, \psi) f(Y_{obs} \mid \theta)$$

we find that we need to work with

$$f(Y_{obs} \mid \theta) = \int f(Y_{obs}, Y_{mis} \mid \theta) \mathrm{d}Y_{mis}.$$

Note that statistical models are usually expressed in the form of a likelihood for the complete data. And so often we find that whilst $f(Y_{obs}, Y_{mis} \mid \theta)$ may have a nice form that is easy to work with, $f(Y_{obs} \mid \theta)$ can be either hard to calculate, or hard to maximize.

In the next section, we will look at a computational approach for maximum likelihood estimation which can be used in missing data problems.

6.2 The Expectation Maximisation (EM) algorithm

The EM (Expectation-Maximisation) algorithm is a technique used for maximising likelihood functions when the data is in some sense incomplete. This can be due to missing data (via any of the mechanisms discussed above), or, it may be that we are able to imagine additional unrecorded information that may

have greatly simplified the analysis if we had known it. We will look at examples of both scenarios.

The EM algorithm uses the following intuitive idea for obtaining parameter estimates:

1. Replace missing values by estimated values using current parameter estimates
2. Re-estimate parameters
3. Repeat steps 1 and 2 until convergence.

6.2.1 Example: Censored exponential distribution continued

Continuing with the censored exponential example from earlier, let's suppose we don't know how to maximise the likelihood function conditional on the missing data:

$$f(Y_{obs}, M \mid \theta) = \theta^{-r} \exp\left(-\frac{1}{\theta}\sum_{i=1}^{r} y_i\right) \exp\left(-\frac{1}{\theta}c(n-r)\right)$$

but that we do know how to maximize the likelihood when there is no missing data[2]. The key idea behind the EM algorithm is to consider what additional data would simplify the analysis.

In this example, we can *augment* the data with the values of the censored observations, $y_{r+1}, \ldots, y_n$. We can carry out this augmentation by calculating the expected values of the censored observations, conditional on θ:

If $Y_j \sim Exp(1/\theta)$, then

$$E(Y_j \mid Y_j > c) = c + \theta, \tag{6.4}$$

due to the memoryless property of the exponential distribution. The EM algorithm in this case can then be summarised as follows:

1. Start with an initial guess $\theta^{(0)}$ of the m.l.e.
2. Set $y_j = c + \theta^{(0)}$ for $j = r+1, \ldots, n$. (expectation)
3. Re-estimate the m.l.e as $\theta^{(1)} = \frac{1}{n}\sum_{i=1}^{n} y_i$. (maximisation)

[2]This is a common situation. Most statistical models were originally developed in part due to their computational tractability.

4. Return to step 2, replacing $\theta^{(0)}$ with $\theta^{(1)}$. Repeat, obtaining $\theta^{(1)}, \theta^{(2)}, \ldots$, until convergence.

We will find that $\lim_{m\to\infty} \theta^{(m)} = \hat{\theta}$, and that this estimate will agree with Equation (6.3).

6.2.2 The general framework

The EM algorithm can be used for a variety of problems, although missing data problems are its primary use. Consequently, slightly different notation will be used here to emphasise the full generality of the approach.

Suppose we observe data $X = x$ and that we wish to maximise the log-likelihood

$$l(\theta) = \log f(x \mid \theta) \tag{6.5}$$

but cannot do so directly. The EM algorithm considers additional data Y (which will usually be missing data in this module), such that it would be easier to maximise the **augmented log-likelihood**

$$\log f(x, y \mid \theta)$$

if we knew that $Y = y$. Of course, we don't know what Y is, so somehow we are going to have to maximise

$$\log f(x, Y \mid \theta)$$

which is a random variable (as Y is a random variable). The way we get round this problem is to take the *expectation* of $\log f(x, Y \mid \theta)$ with respect to Y:

$$\mathbb{E}_Y \left(\log f(x, Y \mid \theta)\right) = \int \log f(x, y \mid \theta) f_Y(y) dy$$

and find θ to maximise this expected log-likelihood. However, without knowing θ, we don't know what the density of Y is. Instead, we simply choose a value θ, let's say $\theta = \theta^{(0)}$, and replace $f_Y(y)$ by $f_{Y|x,\theta^{(0)}}(y \mid \theta^{(0)}, x)$. Ideally, the value $\theta^{(0)}$ will be a guess at the m.l.e of θ in Equation (6.5), perhaps chosen to be physically plausible. Given $\theta^{(0)}$, we then find a new θ by maximising

$$\begin{aligned} Q(\theta \mid \theta^{(0)}) &= \mathbb{E}\left(\log f(x, Y \mid \theta) \mid x, \theta^{(0)}\right) \\ &= \int \log f(x, y \mid \theta) f(y \mid x, \theta^{(0)}) dy. \end{aligned} \tag{6.6}$$

Once we have found θ to maximise Equation (6.6), which we will call $\theta^{(1)}$, we can use this as a new guess for the m.l.e. in Equation (6.5), and start again. The EM algorithm can then be stated as follows:

1. Choose a starting value $\theta^{(0)}$.
2. For $m = 0, 1, \ldots$
   (a) **Expectation**: calculate $Q(\theta \mid \theta^{(m)})$.
   (b) **Maximisation**: choose $\theta^{(m+1)}$ to be the value of θ that maximises $Q(\theta \mid \theta^{(m)})$.

We usually find that $\theta^{(m)}$ quickly converges to a stable answer. We will prove later that

$$f(x \mid \theta^{(m+1)}) \geq f(x \mid \theta^{(m)}),$$

with equality only achieved at a maximum of $f(x \mid \theta)$. Thus, the EM algorithm is a way of finding maxima of $l(\theta)$. Because it is only guaranteed to find local maxima (rather than the global maxima), we usually try several different starting values ($\theta^{(0)}$s) and check they all converge to the same answer.

6.2.3 Example: Censored exponentials continued II

We have data $Y_{obs} = \{y_1, \ldots, y_r\}$, and the information that $Y_{r+1}, \ldots, Y_n$ are censored at c.
We augment the data with $Y_{mis} = \{Y_{r+1}, \ldots, Y_n\}$. Then

$$\log f(Y_{mis}, y_{obs} \mid \theta) = -n \log \theta - \frac{1}{\theta} \left(\sum_{i=1}^{r} y_i + \sum_{i=r+1}^{n} Y_i \right). \quad (6.7)$$

To calculate $Q(\theta \mid \theta^{(m)})$ we must take the expectation of (6.7) with respect to $Y_{mis} = \{Y_{r+1}, \ldots, Y_n\}$, where $Y_{mis,i}$ has density function $f(y \mid y > c, \theta^{(m)})$. Doing so gives

$$\begin{aligned} Q(\theta \mid \theta^{(m)}) &= -n \log \theta - \frac{1}{\theta} \left\{ \sum_{i=1}^{r} y_i + \sum_{i=r+1}^{n} E(Y_i \mid Y_i > c, \theta^{(m)}) \right\} \\ &= -n \log \theta - \frac{1}{\theta} \left\{ \sum_{i=1}^{r} y_i + \sum_{i=r+1}^{n} (c + \theta^{(m)}) \right\}. \end{aligned}$$

The value of θ that maximises this function can then be shown to be

$$\theta^{(m+1)} = \frac{\sum_{i=1}^{r} y_i + (n-r)(c + \theta^{(m)})}{n}. \quad (6.8)$$

It is easy to see that the fixed point of the iteration defined by Equation (6.8) is given by the expression we found earlier (Equation 6.3). Thus, we would not need to use the EM algorithm in this case.

Task: Verify the algebra above.

6.2.4 Convergence

We'll now consider in a little more detail how the EM algorithm works. The algorithm produces a sequence of values $\theta^{(1)}, \theta^{(2)}, \theta^{(3)} \ldots$ which will converge to a local maximum of $L(\theta) = f(x \mid \theta)$.

Theorem 2. *The EM algorithm increases the likelihood at each iteration:*

$$L(\boldsymbol{\theta}^{(m+1)} \mid x) \geq L(\boldsymbol{\theta}^{(m)} \mid x)$$

Proof. By Jensen's inequality[3], for any two densities f and g, we have

$$\begin{aligned}\mathbb{E}_g\left[\log \frac{f(X)}{g(X)}\right] &\leq \log \mathbb{E}_g\left(\frac{f(X)}{g(X)}\right) \\ &= \log \int \frac{f(x)}{g(x)} g(x) \mathrm{d}x \\ &= \log 1 = 0\end{aligned}$$

If we take f to be $f(y \mid x, \theta)$, and g to be $f(y \mid x, \theta^{(m)})$, then

$$\int \log\left(\frac{f(y \mid x, \theta)}{f(y \mid x, \theta^{(m)})}\right) f(y \mid x, \theta^{(m)}) \mathrm{d}y \leq 0. \tag{6.9}$$

Let $\theta = \arg\max Q(\theta \mid \theta^{(m)})$, the maximizer of $Q(\theta \mid \theta^{(m)})$ found in the M-step. Then $Q(\theta \mid \theta^{(m)}) \geq Q(\theta^{(m)} \mid \theta^{(m)})$, and so

$$\begin{aligned}0 &\leq Q(\theta \mid \theta^{(m)}) - Q(\theta^{(m)} \mid \theta^{(m)}) \\ &= \int \log(f(y, x \mid \theta)) f(y \mid x, \theta^{(m)}) \mathrm{d}y - \int \log(f(y, x \mid \theta^{(m)})) f(y \mid x, \theta^{(m)}) \mathrm{d}y \\ &= \int \log\left(\frac{f(x, y \mid \theta)}{f(x, y \mid \theta^{(m)})}\right) f(y \mid x, \theta^{(m)}) \mathrm{d}y. \qquad (6.10)\end{aligned}$$

If we subtract Equation (6.9) from (6.10), we get

$$\int \log\left(\frac{f(x, y \mid \theta) f(y \mid x, \theta^{(m)})}{f(y \mid x, \theta) f(x, y \mid \theta^{(m)})}\right) f(y \mid x, \theta^{(m)}) \mathrm{d}y \geq 0.$$

However, if we consider the terms in the log we have

$$f(x \mid \theta) = \frac{f(x, y|\theta)}{f(y|x, \theta)} \text{ and } \frac{1}{f(x \mid \theta^{(m)})} = \frac{f(y \mid x, \theta^{(m)})}{f(x, y \mid \theta^{(m)})}$$

respectively and these do not depend on y. Hence, we find

$$\log f(x \mid \theta) \geq \log f(x \mid \theta^{(m)})$$

since $\int f(y \mid x, \theta^{(m)}) \mathrm{d}y = 1$. Thus the likelihood in the EM algorithm alway increases as required. ■

[3] Jensen's inequality says that if $h(x)$ is a convex function and X is a random variable, then

$$h(\mathbb{E}X) \leq \mathbb{E}h(X).$$

6.2.5 Exercise: multinomially distributed data

The following example illustrates how the additional data used in the EM algorithm (the Y in the notation above) need not be 'missing data' as such, but can be information that if it were available would make the analysis simpler.

In a dataset, 197 animals are distributed multinomially into four categories. The observed data are given by

$$\mathbf{x} = (x_1, x_2, x_3, x_4) = (125, 18, 20, 34).$$

The probabilities of membership in each category for any animal are given by a genetic model:

$$\left\{\frac{1}{2} + \frac{\pi}{4}, \frac{1}{4}(1-\pi), \frac{1}{4}(1-\pi), \frac{\pi}{4}\right\},$$

for some unknown value of $\pi \in (0, 1)$. The likelihood function for π is then given by

$$f(\mathbf{x} \mid \pi) = \frac{(x_1 + x_2 + x_3 + x_4)!}{x_1! x_2! x_3! x_4!} \left(\frac{2+\pi}{4}\right)^{x_1} \left(\frac{1-\pi}{4}\right)^{x_2} \left(\frac{1-\pi}{4}\right)^{x_3} \left(\frac{\pi}{4}\right)^{x_4}.$$

To maximise this likelihood with the EM algorithm, we again have to consider what 'missing' data might simplify the analysis. In this case, we can convert the likelihood into the more familiar binomial form $\pi^n(1-\pi)^m$ by supposing that there are in fact five categories instead of four, so that the complete data would be given by $\boldsymbol{y} = (y_1, y_2, y_3, y_4, y_5)$, with $x_1 = y_1 + y_2$, $x_2 = y_3$, $x_3 = y_4$ and $x_4 = y_5$. We then suppose that the probabilities for the five new categories are

$$\left\{\frac{1}{2}, \frac{\pi}{4}, \frac{1}{4}(1-\pi), \frac{1}{4}(1-\pi), \frac{\pi}{4}\right\},$$

This gives the likelihood

$$\begin{aligned} f(\boldsymbol{y} \mid \pi) &= \frac{(y_1 + y_2 + y_3 + y_4 + y_5)!}{y_1! y_2! y_3! y_4! y_5} \left(\frac{1}{2}\right)^{y_1} \left(\frac{\pi}{4}\right)^{y_2} \left(\frac{1-\pi}{4}\right)^{y_3} \left(\frac{1-\pi}{4}\right)^{y_4} \left(\frac{\pi}{4}\right)^{y_5} \\ &\propto \pi^{y_2+y_5}(1-\pi)^{y_3+y_4}. \end{aligned}$$

Given a current estimate $\pi^{(m)}$ of the m.l.e., how would you apply one iteration of the EM algorithm to derive an improved estimate?

6.3 EM for exponential family distributions

When the complete data likelihood is a member of the **exponential family**, we will show that the EM algorithm takes on

a simplified form, that involves replacing **sufficient statistics** in the log-likelihood by their expectations (conditional on the current estimate $\theta^{(m)}$).

6.3.1 The exponential family

Let θ be a k dimensional vector $\theta = (\theta_1, \ldots, \theta_k)$. A density function $f(x \mid \theta)$ is said to belong to the **exponential family** if we can write

$$f(x \mid \theta) = \exp\{\sum_{i=1}^{k} A_i(\theta)B_i(x) + C(x) + D(\theta)\}. \qquad (6.11)$$

Many distributions that we commonly work with are members of the exponential family. For example:

- Normal distribution : $\theta = (\mu, \sigma^2)$.

$$\begin{aligned} A_1(\theta) &= \sigma^{-2} \\ A_2(\theta) &= \mu/\sigma^2 \\ B_1(x) &= -x^2/2 \\ B_2(x) &= x \\ C(x) &= 0 \\ D(\theta) &= -(\log(2\pi\sigma^2) + \mu^2/\sigma^2)/2 \end{aligned}$$

- Exponential distribution: $f(x|\theta) = \frac{1}{\theta}\exp(-\frac{x}{\theta})$

$$\begin{aligned} A_1(\theta) &= -\frac{1}{\theta} \\ B_1(x) &= x \\ C(x) &= 0 \\ D(\theta) &= -\log\theta \end{aligned}$$

- Binomial distribution

$$\begin{aligned} A_1(\theta) &= \log\left(\frac{\theta}{1-\theta}\right) \\ B_1(x) &= x \\ C(x) &= \log({}^nC_x) \\ D(\theta) &= n\log(1-\theta) \end{aligned}$$

Gamma, chi-squared, beta, Poisson, Wishart and many other distributions are also members of the exponential family. Examples of distributions that are not in the exponential family include the Student-t distribution, and uniform distributions with uncertain bounds.

We are going to write the density function in a slightly different form to Equation (6.11). The **regular exponential family form** is

$$f(y \mid \phi) = \frac{b(y)\exp\{\phi S(y)\}}{a(\phi)}. \tag{6.12}$$

When written in this form, ϕ is called the **natural** parameter in the distribution of y. ϕ may be a reparameterization of the conventional parameter you are used to working with for that distribution. The statistic $S(y)$ is known as the **natural sufficient statistic**[4] for ϕ. Note that ϕ and $S(y)$ may be vectors.

For example, suppose y has a binomial distribution with probability of success θ. The natural parameter ϕ is then given by

$$\phi = \log\left(\frac{\theta}{1-\theta}\right)$$

and $S(y) = y$. Note also that $b(y) = \binom{n}{y}$ and

$$a(\phi) = \left(1 + e^{\phi}\right)^n.$$

The natural parameter ϕ is typically a non-linear transformation of the **conventional** parameter θ, i.e., $\phi = h(\theta)$, where h is a bijective (one-to-one) function. It is easy to see that if $\hat{\phi}$ maximises $f(y \mid \phi)$, then $\hat{\theta} = h^{-1}(\hat{\phi})$ will maximise $f(y \mid \theta)$. Consequently, it does not matter whether we use the EM algorithm to estimate $\hat{\phi}$ or $\hat{\theta}$.

Lemma 1. *For exponential family distributions in the form given by Equation (6.12),*

$$\frac{\partial \log a(\phi)}{\partial \phi} = \mathbb{E}(S(y) \mid \phi)$$

Proof. Since (6.12) is a density function in y, it must integrate to 1 (when integrated with respect to y), and so

$$a(\phi) = \int b(y)\exp\{\phi S(y)\}\, dy.$$

[4] Recall that a statistic $S(\mathbf{x})$ (a function of data $x_1, \ldots, x_n$) is a sufficient statistic for a parameter θ, if and only if the conditional distribution of $x_1, \ldots, x_n$ given $S(\mathbf{x})$ is independent of θ. Sufficient statistics can be identified using the *factorization theorem*, which states that a statistic $S(\mathbf{x})$ is sufficient for θ if and only if the joint distribution of $X_1, \ldots, X_n$ can be factored as

$$f(x_1, \ldots, x_n \mid \theta) = g(s(\mathbf{x}), \theta)h(x_1, \ldots, x_n), \tag{6.13}$$

where $g(s(\mathbf{x}), \theta)$ depends on $s(\mathbf{x})$ and θ only, and $h(x_1, \ldots, x_n)$ does not depend on θ.

Now differentiate $\log a(\phi)$ with respect to ϕ:

$$\begin{aligned}\frac{\partial}{\partial\phi}\log a(\phi) &= \frac{1}{a(\phi)}\frac{\partial}{\partial\phi}a(\phi)\\ &= \frac{1}{a(\phi)}\int S(y)\,b(y)\exp\{\phi S(y)\}\,dy\\ &= \int S(y)f(y\mid\phi)\,dy\\ &= E\{S(y)\mid\phi\}.\end{aligned}$$

■

Task: Show that for the binomial distribution

$$\log a(\phi) = n\log(1+e^{\phi})$$

and so

$$\frac{d}{d\phi}\log a(\phi) = \frac{ne^{\phi}}{1+e^{\phi}} = n\theta$$

as expected.

More information on exponential families can be found in Section 3.3 of the the MAS376 notes.

6.3.2 EM algorithm for exponential family distributions

Let's now apply the EM algorithm to estimate ϕ. We will assume that the complete data vector (x, Y) is from an exponential family, but again assume that Y is missing. Given all the data (x, Y), i.e. both observed and missing, define $S(x, Y)$ to be the natural sufficient statistic for the unknown parameter ϕ. We need to calculate[5]

$$Q(\phi\mid\phi^{(m)}) = \mathbb{E}\left(\log b(x,Y) + \phi S(x,Y) - \log a(\phi)\mid x,\phi^{(m)}\right).$$

If we differentiate this with respect to ϕ, we find

$$\frac{d}{d\phi}Q(\phi\mid\phi^{(m)}) = \mathbb{E}\left(S(x,Y)\mid x,\phi^{(m)}\right) - \frac{d}{d\phi}\log a(\phi)$$

To solve the maximisation step, we set $\frac{d}{d\phi}Q(\phi\mid\phi^{(m)}) = 0$, which gives

[5]It is important to note the difference between ϕ and $\phi^{(m)}$ here.

$$\mathbb{E}\left(S(x,Y) \mid x, \phi^{(m)}\right) = \frac{d}{d\phi}\log a(\phi)$$
$$= \mathbb{E}(S(x,Y) \mid \phi)$$

If we let[6]

$$S^{(m)} = \mathbb{E}\left(S(x,Y) \mid x, \phi^{(m)}\right)$$

then the M-step of the EM algorithm involves finding ϕ such that

$$\mathbb{E}(S(x,Y) \mid \phi) = S^{(m)}.$$

To summarize, the *EM* algorithm for exponential families is as follows:

1. Start with an initial value $\theta^{(0)}$.
2. For $m = 0, 1, \ldots$ (until convergence)
   (a) **E-step**: Let $S^{(m)} = \mathbb{E}\left(S(x,Y) \mid x, \theta^{(m)}\right)$.
   (b) **M-step**: Choose $\theta^{(m+1)}$ to be the solution θ to the equation
   $$\mathbb{E}\left(S(x,Y) \mid \theta\right) = S^{(m)}.$$

6.3.3 Example: Censored exponential distribution continued III

The complete data (both fully observed and censored observations) is given by $\{\mathbf{x} = (y_1, \ldots, y_r), \boldsymbol{Y} = (y_{r+1}, \ldots, y_n)$. The sufficient statistic for $\phi = \frac{1}{\theta}$ is given by

$$S(\mathbf{x}, \boldsymbol{Y}) = y_1 + \ldots + y_r + Y_{r+1} + \ldots + Y_n.$$

We begin with an initial guess $\theta^{(m)}$ and compute (it doesn't matter if we work with θ or $\phi = \frac{1}{\theta}$ here)

$$\begin{aligned} S^{(m)} &= \mathbb{E}\left(S(x,Y) \mid y_1, \ldots, y_r, Y_{r+1} \geq c, \ldots, Y_n \geq c, \theta^{(m)}\right) \\ &= y_1 + \ldots, y_r + c + \theta^{(m)} + \ldots + c + \theta^{(m)} \\ &= \sum_{i=1}^{r} y_i + (n-r)(c + \theta^{(m)}). \end{aligned}$$

In computing $S^{(m)}$, we can then see that we have simply plugged in the expectations of the missing values, conditional on $\theta^{(m)}$, into the expression for the sufficient statistic $S(\mathbf{x}, \boldsymbol{Y})$. Now,

$$\mathbb{E}(S(\mathbf{x}, \boldsymbol{Y}) \mid \theta) = \theta + \ldots + \theta = n\theta.$$

[6] Note the difference between $\mathbb{E}(S(x,Y) \mid \phi)$ and $\mathbb{E}(S(x,Y) \mid x, \phi)$!

Finally, we find $\theta^{(m+1)}$ by solving

$$E\{S(\mathbf{x}, \boldsymbol{Y}) \mid \theta\} = S^{(m)},$$

i.e., by solving

$$n\theta = \sum_{i=1}^{r} y_i + (n-r)(c + \theta^{(m)}) \tag{6.14}$$

which gives

$$\theta^{(m+1)} = \frac{\sum_{i=1}^{r} y_i + (n-r)(c + \theta^{(m)})}{n} \tag{6.15}$$

which is the same as we found in Equation (6.8).

6.4 Example: Mixture distributions

The distribution of some random variable X in a population can be represented by a *mixture distribution*, when the population can be split into subgroups, with each subgroup having a different distribution for X. For example, the distribution of heights of twenty-year olds could be modelled as a mixture distribution; a combination of two distributions, one representing males and one representing females. In ANOVA type scenarios, we may for example have data $x_{i,j}$ corresponding to the jth member of the ith group. A fixed effects model for the data would then be

$$x_{i,j} = \mu_i + \varepsilon_{i,j}, \tag{6.16}$$

with $\varepsilon_{i,j} \sim N(0, \sigma^2)$.

Example: mixture of two normal distributions

Let $X_1, \ldots, X_n$ be independent observations drawn from a mixture of $N(\mu, \sigma^2)$ with probability ϕ, and $N(\nu, \tau^2)$ with probability $1 - \phi$. Both variances σ^2 and τ^2 are known. Denote the observed values of $X_1, \ldots, X_n$ by $\mathbf{x} = (x_1, \ldots, x_n)$. We will use the EM algorithm to obtain the m.l.e. of μ, ν and ϕ.

The first question to ask is, what additional data would simplify the analysis? In this case, knowing which normal distribution each observation X_i came from.

For each observation x_i we introduce an indicator variable y_i, where y_i identifies which normal distribution x_i was actually drawn from:

$$y_i = \begin{cases} 1 & \text{if } X_i \text{ drawn from } N(\mu, \sigma^2) \\ 0 & \text{if } X_i \text{ drawn from } N(\nu, \tau^2). \end{cases} \tag{6.17}$$

We will first derive the complete data likelihood function. We need to consider the joint density of $\mathbf{x}$ and $\boldsymbol{y}$. Note that

$$f(\mathbf{x}, \boldsymbol{y} \mid \phi, \mu, \nu) = f(\mathbf{x} \mid \boldsymbol{y}, \phi, \mu, \nu) f(\boldsymbol{y} \mid \phi, \mu, \nu). \tag{6.18}$$

The two densities on the RHS are easier to work with: $f(\boldsymbol{y} \mid \phi, \mu, \nu)$ will be in the form of the standard binomial likelihood function; and conditional on $\boldsymbol{y}$, we know which of the two normal distributions each element of $\mathbf{x}$ comes from, so $f(\mathbf{x} \mid \boldsymbol{y}, \phi, \mu, \nu)$ will simply be a product of normal density functions. We have[7]

$$\begin{aligned} f(\boldsymbol{y} \mid \phi, \mu, \nu) &= \phi^{\sum y_i}(1-\phi)^{n-\sum y_i} \\ f(\mathbf{x} \mid \boldsymbol{y}, \phi, \mu, \nu) &\propto \exp\left\{ -\frac{1}{2\sigma^2} \sum_{i:y_i=1} (x_i - \mu)^2 - \frac{1}{2\tau^2} \sum_{i:y_i=0} (x_i - \nu)^2 \right\}, \end{aligned}$$

so the complete data log-likelihood is given by

$$\begin{aligned} \log f(\mathbf{x}, \boldsymbol{y} \mid \phi, \mu, \nu) &= K + \sum y_i \log \phi + (n - \sum y_i) \log(1-\phi) - \frac{1}{2\sigma^2} \sum_{i:y_i=1} (x_i - \mu)^2 \\ &\quad - \frac{1}{2\tau^2} \sum_{i:y_i=0} (x_i - \nu)^2, \end{aligned}$$

for some constant K. This density is a member of the exponential family and the sufficient statistics for $\theta = (\phi, \mu, \nu)$ are

$$S = \left(\sum_i y_i, \sum_{i:y_i=1} x_i, \sum_{i:y_i=0} x_i \right),$$

which follows from the factorization theorem. We can write

$$\begin{aligned} \log f(\mathbf{x}, \boldsymbol{y} \mid \phi, \mu, \nu) &= K + \sum y_i \log \phi + (n - \sum y_i) \log(1-\phi) - \frac{1}{2\sigma^2} \sum_{i:y_i=1} (\mu^2 - 2\mu x_i) \\ &\quad - \frac{1}{2\tau^2} \sum_{i:y_i=0} (\nu^2 - 2\nu x_i) - \frac{1}{2\sigma^2} \sum_{i:y_i=1} x_i^2 - \frac{1}{2\tau^2} \sum_{i:y_i=0} x_i^2. \end{aligned}$$

Given the current parameter estimate $\theta^{(m)} = (\phi^{(m)}, \mu^{(m)}, \nu^{(m)})$, we must calculate

$$S^{(m)} = \mathbb{E}\left(S(\mathbf{x}, \boldsymbol{Y}) \mid \mathbf{x}, \theta^{(m)} \right).$$

Consider first $\mathbb{E}(Y_i \mid x_i, \theta^{(m)})$. We have

$$\begin{aligned} \mathbb{E}(Y_i \mid x_i, \theta^{(m)}) &= \mathbb{P}(Y_i = 1 \mid x_i, \theta^{(m)}) \\ &= \frac{\mathbb{P}(Y_i = 1 \mid \theta^{(m)}) f(x_i \mid Y_i = 1, \theta^{(m)})}{f(x_i \mid \theta^{(m)})} \end{aligned} \tag{6.19}$$

[7] Task: Why is there no $\binom{n}{\sum y_i}$ term here?

from Bayes' theorem. Then

$$\begin{aligned}\mathbb{P}(Y_i = 1 \mid \theta^{(m)}) &= \phi^{(m)} \\ f(x_i \mid Y_i = 1, \theta^{(m)}) &= \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{1}{2\sigma^2}(x_i - \mu^{(m)})^2\right\}.\end{aligned}$$

Deriving the density $f(x_i \mid \theta^{(m)})$ is a little harder. We again use the same conditioning idea as in Equation (6.18): conditional on the value of Y_i, we know the density of x_i. We write

$$f(x_i \mid \theta^{(m)}) = f(x_i \mid Y_i = 1, \theta^{(m)})\mathbb{P}(Y_i = 1 \mid \theta^{(m)}) + f(x_i \mid Y_i = 0, \theta^{(m)})\mathbb{P}(Y_i = 0 \mid \theta^{(m)}),$$

i.e., we integrate out Y_i from the joint distribution of x_i and Y_i. We can now write

$$\begin{aligned}f(x_i \mid \theta^{(m)}) &= \phi^{(m)} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{1}{2\sigma^2}(x_i - \mu^{(m)})^2\right\} \\ &\quad + (1 - \phi^{(m)}) \frac{1}{\sqrt{2\pi\tau^2}} \exp\left\{-\frac{1}{2\tau^2}(x_i - \nu^{(m)})^2\right\}\end{aligned}$$

and we write that

$$\mathbb{P}(Y_i = 1 \mid x_i, \theta^{(m)}) = p_i^{(m)},$$

where $p_i^{(m)}$ is obtained by substituting these results into (6.19). Now we need to find

$$\mathbb{E}\left(\sum_{i:Y_i=1} x_i \mid \mathbf{x}, \theta^{(m)}\right).$$

Firstly, note that

$$\sum_{i:Y_i=1} x_i = \sum_{i=1}^{n} Y_i x_i.$$

Then

$$\begin{aligned}\mathbb{E}(Y_i x_i \mid x_i, \theta^{(m)}) &= x_i \mathbb{E}(Y_i \mid x_i, \theta^{(m)}) \\ &= p_i^{(m)} x_i\end{aligned}$$

Additionally,

$$\sum_{i:Y_i=0} x_i = \sum_{i=1}^{n} (1 - Y_i) x_i, \tag{6.20}$$

and so

$$\begin{aligned}\mathbb{E}((1 - Y_i)x_i \mid x_i, \theta^{(m)}) &= x_i \mathbb{E}((1 - Y_i) \mid x_i, \theta^{(m)}) \\ &= (1 - p_i^{(m)}) x_i\end{aligned}$$

We now have the expected value of the sufficient statistics conditional on $\theta^{(m)}$ and $X = \mathbf{x}$:

$$S^{(m)} = (\sum_i p_i^{(m)}, \sum_i p_i^{(m)} x_i, \sum_i (1 - p_i^{(m)}) x_i).$$

Next, we need to derive

$$\mathbb{E}(S(x, Y) \mid \theta).$$

Firstly,

$$\mathbb{E}\left(\sum_i Y_i \mid \theta\right) = n\phi.$$

Also,

$$\begin{aligned}
\mathbb{E}(Y_i x_i \mid \theta) &= \int 1.x.\mathbb{P}(Y_i = 1 \mid \theta) f(x \mid \theta, Y_i = 1) dx \\
&\quad + \int 0.x.\mathbb{P}(Y_i = 0 \mid \theta) f(x \mid \theta, Y_i = 0) dx \\
&= \phi \int x f(x \mid \theta, Y_i = 1) dx \\
&= \phi\mu
\end{aligned}$$

Similarly,

$$E\{(1 - Y_i)x_i \mid \theta\} = (1 - \phi)\nu$$

This gives us

$$\begin{aligned}
n\phi &= \sum p_i^{(m)} \\
n\phi\mu &= \sum p_i^{(m)} x_i \\
n(1 - \phi)\nu &= \sum (1 - p_i^{(m)}) x_i.
\end{aligned}$$

So, for $\theta^{(m+1)}$ we have

$$\begin{aligned}
\phi^{(m+1)} &= \frac{\sum p_i^{(m)}}{n} \\
\mu^{(m+1)} &= \frac{\sum p_i^{(m)} x_i}{\sum p_i^{(m)}} \\
\nu^{(m+1)} &= \frac{\sum (1 - p_i^{(m)}) x_i}{\sum (1 - p_i^{(m)})}.
\end{aligned}$$

See the R code for an implementation of this algorithm. Finally, note that this situation is similar to a fixed effects model where we had lost the information about which group each observation belonged to.

6.4.1 Exercise

Data $X_1, \ldots, X_n$ are independent and identically distributed, with the distribution of each X_i being a mixture of Poisson distributions, i.e.,

$$\begin{aligned}
X_i &\sim Poisson(\lambda) \text{ with probability } p, \\
X_i &\sim Poisson(\gamma) \text{ with probability } 1 - p
\end{aligned}$$

The parameters λ, γ and p are all unknown, and for any X_i, it is not known which of the two Poisson distributions X_i was actually sampled from. Given initial guesses for the maximum likelihood estimates of λ, γ and p given $X_1, \ldots, X_n$, applying one iteration of the EM algorithm, what are the new estimates of the maximum likelihood estimates of λ, γ and p?

6.5 Multivariate normal: data with ignorable missing data mechanism

Consider a k-dimensional multivariate normal distribution

$$\boldsymbol{Y} \sim N_k(\boldsymbol{\mu}, \boldsymbol{\Sigma}).$$

Suppose we are given observations $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n$. Then it is easy to see that the sufficient statistics for $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ are

$$\sum_{i=1}^{n} \boldsymbol{y}_i \quad \text{and} \quad \sum_{i=1}^{n} \boldsymbol{y}_i \boldsymbol{y}_i^\top,$$

which are the multi-dimensional versions of the sample first (mean) and second moments. The maximum likelihood estimator of $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ are

$$\begin{aligned}
\hat{\boldsymbol{\mu}} &= \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{y}_i \\
&= \bar{\boldsymbol{y}} \\
\hat{\boldsymbol{\Sigma}} &= \frac{1}{n} \sum_{i=1}^{n} (\boldsymbol{y}_i - \hat{\boldsymbol{\mu}})(\boldsymbol{y}_i - \hat{\boldsymbol{\mu}})^\top \\
&= \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{y}_i \boldsymbol{y}_i^\top - \hat{\boldsymbol{\mu}} \hat{\boldsymbol{\mu}}^\top
\end{aligned}$$

which are the sample mean and covariance matrix (using n rather than $n-1$ as the denominator), and which can be seen to be functions of the sufficient statistics.

Now suppose there are missing data, with an ignorable (MCAR or MAR) mechanism. Let $\boldsymbol{y}_{mis,i}$ and $\boldsymbol{y}_{obs,i}$ denote the observed and missing parts of $\boldsymbol{y}_i$. We will allow different parts of each $\boldsymbol{y}_i$ to be missing.

The EM algorithm for exponential families (and the multivariate normal is a member of the exponential family) tells us we only need consider the expectation of the sufficient statistics. We need to

1. Start with an initial value $\theta^{(0)} = (\boldsymbol{\mu}^{(0)}, \boldsymbol{\Sigma}^{(0)})$ which we can estimate, e.g., by complete case analysis

2. Let

$$\mathbf{S}_1^{(m)} = \mathbb{E}\left(\sum_{i=1}^{n} \boldsymbol{y}_i \mid \boldsymbol{y}_{obs}, \theta^{(m)}\right) \qquad \mathbf{S}_2^{(m)} = \mathbb{E}\left(\sum_{i=1}^{n} \boldsymbol{y}_i \boldsymbol{y}_i^\top \mid \boldsymbol{y}_{obs}, \theta^{(m)}\right)$$

where $\theta^{(m)} = (\boldsymbol{\mu}^{(m)}, \boldsymbol{\Sigma}^{(m)})$ denotes the parameter estimates obtained in iteration m.

- Let's first evaluate $\mathbb{E}(\boldsymbol{y} \mid \theta^{(m)}, \boldsymbol{y}_{obs})$ and $\mathbb{E}(\boldsymbol{y}\boldsymbol{y}^\top \mid \theta^{(m)}, \boldsymbol{y}_{obs})$, To do this, if we write[8]

$$\boldsymbol{y}_i = \begin{pmatrix} \boldsymbol{y}_{obs,i} \\ \boldsymbol{y}_{mis,i} \end{pmatrix} \text{ and } \boldsymbol{\mu} = \begin{pmatrix} \boldsymbol{\mu}_{obs} \\ \boldsymbol{\mu}_{mis} \end{pmatrix} \text{ and } \boldsymbol{\Sigma} = \begin{pmatrix} \boldsymbol{\Sigma}_{obs,obs} & \boldsymbol{\Sigma}_{mis,obs} \\ \boldsymbol{\Sigma}_{mis,obs} & \boldsymbol{\Sigma}_{mis,mis} \end{pmatrix}$$

then clearly

$$\mathbb{E}(\boldsymbol{y}_{obs} \mid \theta^{(m)}, \boldsymbol{y}_{obs}) = \boldsymbol{y}_{obs} \text{ and } \mathbb{E}(\boldsymbol{y}_{obs}\boldsymbol{y}_{obs}^\top \mid \theta^{(m)}, \boldsymbol{y}_{obs}) = \boldsymbol{y}_{obs}\boldsymbol{y}_{obs}^\top.$$

- To find $\mathbb{E}(\boldsymbol{y}_{mis} \mid \theta^{(m)}, \boldsymbol{y}_{obs})$ we need the conditional multivariate Gaussian formulae which are that

$$\begin{aligned} \mathbb{E}(\boldsymbol{y}_{mis} \mid \theta, \boldsymbol{y}_{obs}) &= \boldsymbol{\mu}_{mis} + \boldsymbol{\Sigma}_{mis,obs}\boldsymbol{\Sigma}_{obs,obs}^{-1}(\boldsymbol{y}_{obs,i} - \boldsymbol{\mu}_{obs}) \\ &= \mathbf{m}_{mis} \text{ say} \\ \mathbb{V}\text{ar}(\boldsymbol{y}_{mis} \mid \theta, \boldsymbol{y}_{obs}) &= \boldsymbol{\Sigma}_{mis,mis} - \boldsymbol{\Sigma}_{mis,obs}\boldsymbol{\Sigma}_{obs,obs}^{-1}\boldsymbol{\Sigma}_{obs,mis} \end{aligned}$$

- Then

$$\mathbb{E}(\boldsymbol{y} \mid \theta^{(m)}, \boldsymbol{y}_{obs}) = \begin{pmatrix} \boldsymbol{y}_{obs} \\ \mathbf{m}_{mis} \end{pmatrix}$$

and we can calculate

$$\mathbb{E}\left(\sum_{i=1}^{n} \boldsymbol{y} \mid \theta^{(m)}, \boldsymbol{y}_{obs}\right) =: \mathbf{S}_1^{(m)}.$$

- Similarly

$$\mathbb{V}\text{ar}\left(\begin{pmatrix} \boldsymbol{y}_{obs} \\ \boldsymbol{y}_{mis} \end{pmatrix} \mid \theta, \boldsymbol{y}_{obs}\right) = \begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbb{V}\text{ar}(\boldsymbol{y}_{mis} \mid \theta, \boldsymbol{y}_{obs}) \end{pmatrix}.$$

We can then calculate

$$\mathbf{S}_2^{(m)} := \sum_i \mathbb{E}(\boldsymbol{y}_i \boldsymbol{y}_i^\top \mid \theta, \boldsymbol{y}_{obs}) = \sum_i \mathbb{V}\text{ar}(\boldsymbol{y}_i \mid \theta, \boldsymbol{y}_{obs}) + \sum_i \mathbb{E}(\boldsymbol{y}_i \mid \theta^{(m)}, \boldsymbol{y}_{obs})\mathbb{E}(\boldsymbol{y}_i \mid \theta^{(m)}, \boldsymbol{y}_{obs})^\top.$$

[8] Note that I've ordered the elements of $\boldsymbol{y}$ here so that all the observed elements appear first. In practice, we don't need to this (see the R code) - it just makes the presentation simpler

3. Finally, we have to choose $\theta^{(m+1)}$ to be the solution to

$$\mathbf{S}_1^{(m)} = \mathbb{E}(\mathbf{S}_1 \mid \theta^{(m+1)})$$
$$\mathbf{S}_2^{(m)} = \mathbb{E}(\mathbf{S}_2 \mid \theta^{(m+1)})$$

which implies that

$$\mathbf{S}_1^{(m)} = n\boldsymbol{\mu}^{(m+1)}$$
$$\mathbf{S}_2^{(m)} = n\left(\boldsymbol{\Sigma}^{(m+1)} + \boldsymbol{\mu}^{(m+1)}\boldsymbol{\mu}^{(m+1)\top}\right)$$

and hence

$$\mu^{(m+1)} = \frac{1}{n}\mathbf{S}_1^{(m)}$$
$$\boldsymbol{\Sigma}^{(m+1)} = \frac{\mathbf{S}_2^{(m)}}{n} - \boldsymbol{\mu}^{(m+1)}\boldsymbol{\mu}^{(m+1)\top}.$$

The EM algorithm then works by iterating between estimating $\mathbf{S}_1^{(m)}, \mathbf{S}_2^{(m)}$, and $\boldsymbol{\mu}^{(m+1)}, \boldsymbol{\Sigma}^{(m+1)}$

See the R code on MOLE for a numerical demonstration.

6.6 Relevance to regression

How does this apply to regression problems where we want to predict y from $x_1, \ldots, x_p$? If we assume multivariate normality for $(y, x_1, \ldots, x_k)$ then can use the example above to find the maximum likelihood estimates for regression of y on X (see p239 in Little and Rubin). If any of the covariates are categorical (factors) then we need to develop an alternative model for the joint specification of the multivariate distribution. However, the calculation for all of these approaches is very involved and difficult to implement. Thankfully, there is an alternative approach which is much simpler to apply and understand, which works almost as well in most situations (and can even work better in some). This is the topic of the next chapter.

Chapter 7

Missing data: Multiple imputation

Likelihood based approaches to missing data, such as the EM algorithm, are hard because i) a coherent joint statistical model needs to be proposed to describe the entire dataset, and ii) the calculations involved in finding the MLE can be challenging. This motivated statisticians to look for alternative approaches which allow practitioners to apply standard complete-data methods using off-the-shelf software.

In Chapter 5 we saw the single imputation approach, in which, if the data recorded in case i was $y_{i1}, \ldots, y_{ik}$, then we would fill in missing values in y_{i1} by building a regression model to predict it from $y_{i2}, \ldots, y_{ik}$. We listed three problems with this approach:

1. It only uses the mean of the regression model prediction, ignoring the random error part.
2. It assumes the estimated parameter values are known without error
3. It relies on complete cases only to estimate the regression coefficients

The first two problems mean that we underestimate uncertainty, whereas the third means that we can over-estimate the uncertainty.

The idea in **multiple imputation** is to create multiple imputed datasets (using a stochastic imputation approach), to analyse them all using the standard complete case analysis, and then to combine the results in some sensible way. So if Y_{mis} is the missing data, then our approach would be to propose multiple

complete datasets

$$\begin{aligned} Y^{(1)} &= (Y_{obs}, Y_{mis}^{(1)}) \\ &\vdots \\ Y^{(m)} &= (Y_{obs}, Y_{mis}^{(m)}) \end{aligned}$$

where each $Y_{mis}^{(i)}$ is an imputed value of the missing data (we'll discuss how to do this later). If θ is the quantity of interest in our statistical analysis (e.g. regression coefficients, a mean vector, etc), then we will analyse each complete dataset to obtain m estimates, $\hat{\theta}^{(1)}, \ldots, \hat{\theta}^{(m)}$, as well as m variance-covariance matrices $V^{(1)}, \ldots, V^{(m)}$ corresponding to each estimate. Note that each of these analyses is applied to a complete dataset, ignoring the fact that the missing data has been imputed. Because it is a 'complete' dataset, the statistical method for doing this is usually simple (e.g., estimating a mean or variance, fitting a regression model, a time-series analysis, etc) and can be carried out using standard approaches and software. We then combine these different complete data estimates to provide statistical inference for θ that takes into account the fact that some of the data is missing.

Note that by applying a complete-data analysis to an imputed dataset, we will not obtain the same inference we would have if the dataset had no missing values. Missing data is lost information that we cannot hope to retrieve. However, what we can hope for is that we achieve valid statistical inference using complete-data analysis methods on the imputed datasets.

Finally, note that the philosophy behind the multiple imputation (MI) approach is quite different to the thinking behind likelihood based approaches (such as EM). There, we specified a full multivariate model of the data, and analysed it using a likelihood based approach using the multivariate density function $f(y|\theta)$. The EM algorithm then tells us how to proceed with a maximum likelihood analysis. In MI, imputed datasets are created, often using models that bear no relation to $f(y|\theta)$. We specify a multivariate imputation method on a variable-by-variable basis using a set of conditional densities, one for each variable. There is no requirement for this even to determine a valid multivariate distribution for the data.

7.1 Combining estimates

Let's briefly consider a Bayesian approach to dealing with missing data. The Bayesian approach to statistics is to describe probability distributions for all unknown quantities, and then to calculate posterior distributions by conditioning upon the observed information. In the notation of Chapter 5, we would try to find

$$\begin{aligned} f(\theta \mid Y_{obs}) &= \int f(\theta, Y_{mis} \mid Y_{obs}) \mathrm{d}Y_{mis} \\ &= \int f(\theta \mid Y_{mis}, Y_{obs}) f(Y_{mis} \mid Y_{obs}) \mathrm{d}Y_{mis}. \end{aligned} \tag{7.1}$$

We aren't going to use a Bayesian approach in this module, but we are going to use Equation (7.1) to derive a sensible approach for combining estimates by considering what the posterior mean and variance of θ would be. Multiple imputation approximates this posterior by

1. Drawing $Y_{mis}^{(i)} \sim f(Y_{mis} \mid Y_{obs})$, for $i = 1, \ldots m$.
2. Using the approximation[1]

$$f(\theta \mid Y_{obs}) \approx \frac{1}{m} \sum_{i=1}^{m} f(\theta \mid Y_{mis}^{(m)}, Y_{obs}).$$

If we summarize this posterior distribution by its mean and variance (and we know that as the number of data points grows large the posterior will be approximately normal), then

$$\mathbb{E}(\theta \mid Y_{obs}) = \mathbb{E}\left(\mathbb{E}(\theta \mid Y_{mis}, Y_{obs}) \mid Y_{obs}\right)$$

by the tower property of expectation[2]. Similarly, by the law of total variance[3]

$$\mathbb{V}\mathrm{ar}(\theta \mid Y_{obs}) = \mathbb{E}\left(\mathbb{V}\mathrm{ar}(\theta \mid Y_{mis}, Y_{obs}) \mid Y_{obs}\right) + \mathbb{V}\mathrm{ar}\left(\mathbb{E}(\theta \mid Y_{mis}, Y_{obs}) \mid Y_{obs}\right).$$

Using the multiple imputation approximation we get

$$\begin{aligned} \mathbb{E}(\theta \mid Y_{obs}) &\approx \frac{1}{m} \sum_{i=1}^{m} \mathbb{E}(\theta | Y_{mis}^{(m)}, Y_{obs}) \\ &= \frac{1}{m} \sum_{i=1}^{m} \hat{\theta}^{(i)} = \bar{\theta} \end{aligned} \tag{7.2}$$

[1] Note that $f(\theta | Y_{mis}^{(m)}, Y_{obs})$ is the posterior distribution based on complete data. Many of the Bayesian methods that are easy to apply to complete data are hard to implement on missing data problems.

[2] For any two random variables X and Y we can show that $\mathbb{E}X = \mathbb{E}(\mathbb{E}(X|Y))$.

[3] For any two random variables X and Y we can show that $\mathbb{V}\mathrm{ar}(X) = \mathbb{E}(\mathbb{V}\mathrm{ar}(X \mid Y)) + \mathbb{V}\mathrm{ar}(\mathbb{E}(X \mid Y))$.

where each $\hat{\theta}^{(i)} = \mathbb{E}(\theta|Y_{mis}^{(i)}, Y_{obs})$ is the estimate from analysing the i^{th} imputed dataset. Similarly, we can estimate the variance-covariance matrix for the estimator from the mean of the variance-covariance estimates from analysing each imputed dataset analysis using

$$\begin{aligned}\mathbb{V}\text{ar}(\theta \mid Y_{obs}) &\approx \frac{1}{m}\sum_{i=1}^{m} V^{(i)} + \frac{1}{m-1}\sum_{i=1}^{m}(\hat{\theta}^{(i)} - \bar{\theta})^2 \\ &= \bar{V} + B\end{aligned}$$

where $V^{(i)} = \mathbb{V}\text{ar}(\theta \mid Y_{mis}^{(i)}, Y_{obs})$ is the variance of the estimator from analysing the i^{th} imputed dataset. Here $\bar{V}$ represents the average within-imputation variability, and B is the between-imputation variability. Note that when the number of imputed datasets used (m) is small, a slightly improved estimator of the posterior variance is usually used:

$$\mathbb{V}\text{ar}(\theta \mid Y_{obs}) \approx \bar{V} + (1 + \frac{1}{m})B. \tag{7.3}$$

It is often useful to calculate the proportion of the total variance that is attributable to the missing data, which is the ratio of the estimated between-imputation variance and the total variance

$$\gamma_m = \frac{(1 + \frac{1}{m})B}{\bar{V} + (1 + \frac{1}{m})B},$$

as this represents the fraction of missing information. It is usually calculated separately for each parameter.

7.1.1 Simple example

Let's now look at a simple example to see how this is done. Code corresponding to this example is available on MOLE. Let's suppose that we are given data

```
> head(data.mis)
           Y        X1          X2
1  2.8602873  1.782878  0.02354324
2  5.7709612  2.398403  2.52554504
3  0.2499038 -2.684407  1.06947895
4  2.7141969 -0.118882  0.61551215
5 -3.6779760        NA -1.67041715
6 -2.9989075  1.591456 -2.26128687
```

and that we wish to build a regression model to predict Y from X1 and X2. Missing values occur in the X1 variables only in this dataset. By default, the following R command will perform the complete case analysis (i.e., it will disregard row 5 in the data)

```
> lm(Y~X1+X2, data.mis)

Call:
lm(formula = Y ~ X1 + X2, data = data.mis)

Coefficients:
(Intercept)           X1           X2
     0.1194       0.1260       2.2157
```

Let's use a very simple imputation method, and randomly select from the observed values of X1 to fill in any missing gaps. We can do this with the following function

```
Impute.data <- function(data.mis){
  X1.imp <- data.mis$X1
  X1.imp[is.na(X1)] <- sample(X1[!is.na(X1)], size = sum(is.na(X1)), replace=TRUE)
  data.impute <- data.mis
  data.impute$X1 <- X1.imp
  return(data.impute)
}
```

Note that this method of imputation is not necessarily a good idea as there may be a relationship between $X2$, Y and $X1$ that we are ignoring by randomly selecting observed values of $X1$, but we use it here for illustrative purposes as it is simple to explain. To perform multiple imputation, we need to create several imputed datasets, and analyse each of them.

```
> m <- 30
> fit.imp <- list()
> theta <- matrix(nr=m, nc=3)
> V <- matrix(nr=m, nc=3)
> # we'll just store the diagonal terms in the variance-covariance matrices.
> for(i in 1:m){
+   data.imp <- Impute.data(data.mis)
+   fit <- lm(Y~X1+X2, data = data.imp)
+   theta[i,] <- coef(fit)
+   V[i,] <- diag(vcov(fit))
+ }
>
> head(theta)
              [,1]       [,2]     [,3]
[1,] -0.0139437917 0.11500334 2.238993
[2,]  0.0088226384 0.11320626 2.240407
[3,] -0.0012812529 0.06523535 2.263506
[4,] -0.0009265703 0.08110167 2.257466
```

```
[5,] -0.0092877747 0.07383699 2.255400
[6,]  0.0193935329 0.01964498 2.280595
>
>
```

Finally, we can pool these estimates using Equations (7.2) and (7.3).

```
> (theta.hat <- colMeans(theta))
[1] 0.01280015 0.08765989 2.24967282
> (var.theta <- colMeans(V) + (1+1/m)* diag(cov(theta)))
[1] 0.032715996 0.008667900 0.005380848
> sqrt(var.theta)
[1] 0.18087564 0.09310156 0.07335427
> (proportion_of_var_due_to_missingness <- (1+1/m)* diag(cov(theta))/var.theta)
(Intercept)          X1          X2
0.008068461 0.214364481 0.089528838
```

Thankfully, we don't need to do this ourselves most of the time. There is an R package called `mice`, which does most of the work.

```
> library(mice)
> data.mice <- mice(data.mis, m=m, method='sample', seed=1)
> fit.mice <- with(data.mice, lm(Y~X1+X2))
> (fit.mice.pool <- pool(fit.mice))
Call: pool(object = fit.mice)

Pooled coefficients:
(Intercept)          X1          X2
 0.01295369  0.08634204  2.25011920

Fraction of information about the coefficients missing due to nonresponse:
(Intercept)          X1          X2
 0.07925861  0.28971995  0.16058329
> round(summary(fit.mice.pool),2)
             est   se     t    df Pr(>|t|) lo 95 hi 95 nmis  fmi lambda
(Intercept) 0.01 0.18  0.07 24.99     0.94 -0.36  0.39   NA 0.08   0.01
X1          0.09 0.09  0.92 19.63     0.37 -0.11  0.28    5 0.29   0.22
X2          2.25 0.07 30.60 22.92     0.00  2.10  2.40    0 0.16   0.09
```

Notice that these answers differ to ours, but not by much. This is because this is a stochastic procedure, and so each time we run the analysis we will get a slightly different answer. Note also that usually, a far smaller number of imputed datasets is used ($m = 10$ is not uncommon), and so the variability will be even

greater in those cases. Typically, the higher the proportion of missing data there is, the larger the number of imputed datasets it will be necessary to analyse.

Note that `mice` has a number of other nice features. For example, `cc(data)` returns all the complete cases (i.e. all the complete rows in `data`), `cci(data)` returns an indicator of which rows are complete. Then `md.pattern` and `md.pairs` give information on the pattern of missing data. For example,

```
> md.pattern(data.mis)
   Y X1 X2
20 1  1  1  0
 3 1  0  1  1
 5 1  1  0  1
 2 1  0  0  2
   0  5  7 12
```

which tells us there are 20 complete rows, 3 rows where only X1 is missing, 5 rows where only X2 is missing, a total of 7 rows where X2 is missing etc.

There are a number of nice tutorials on the `mice` package available on their github page at https://github.com/stefvanbuuren/mice

7.2 Imputation methods

We split the imputation approach up into three steps:

1. create multiple imputed datasets $Y^{(1)}, \ldots, Y^{(m)}$
2. analyse each with a standard complete data method
3. pool the results using Equations (7.2) and (7.3)

Figure 7.1 shows the work flow and gives the main commands used in the `mice` R package.

The first step is to create multiple imputed datasets. The theory above suggests that we need to draw missing values from the conditional distribution given the observed data:

$$Y_{mis}^{(i)} \sim f(Y_{mis} \mid Y_{obs}).$$

This is often difficult, because we need to integrate over the

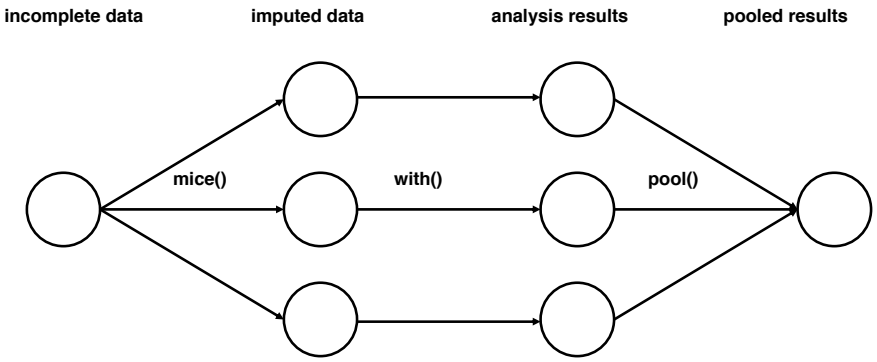

Figure 7.1: Visual illustration of the work flow in a multiple imputation analysis. Figure taken from van Buuren and Groothuis-Oudshoorn 2011

imputation parameters[4] β

$$f(Y_{mis} \mid Y_{obs}) = \int f(\beta, Y_{mis} \mid Y_{obs}) \mathrm{d}\beta$$

As this is difficult (see the next section), sometimes we use simpler methods that approximate draws from this distribution. The simplest approach is to use what Little and Rubin call **Improper MI**, and draw

$$Y_{mis}^{(i)} \sim f(Y_{mis} \mid Y_{obs}, \tilde{\beta})$$

where $\tilde{\beta}$ is an estimate of β obtained using some other method, such as in a complete case analysis. This can work reasonably when there is only a small fraction of missing data, but in general, it does not provide valid inferences as it does not propagate the uncertainty that arises from estimating β.

Another approach is to first draw a value of β from its posterior distribution based upon a subset of the data such as all the complete cases, Y_{obs-cc} (where the posterior distribution is typically easy to calculate), and then to draw Y_{mis} based on this value, i.e.,

$$\begin{aligned} \tilde{\beta}^{(i)} &\sim f(\beta | Y_{obs-cc}) \\ Y_{mis}^{(i)} &\sim f(Y_{mis} \mid Y_{obs}, \tilde{\beta}^{(i)}). \end{aligned}$$

This approach propagates uncertainty about β, but does not use all of the information to draw the value of β as it uses only the complete cases, and so can over estimate the uncertainty.

[4] I have used β instead of θ here to distinguish between the parameters used in the imputation model (β), and the statistical quantity of interest (θ). In practice, some of the β parameters may also be of interest (i.e. part of θ).

An approach that solves both of these problems, and which has become popular in the past decade, is known as Multiple Imputation with Chained Equations (MICE).

7.2.1 Multiple imputation with chained equations (MICE)

One of the problems we can find when doing multiple imputation is that for a given Y_j, we may find that some of the predictors $Y_{-j} = (Y_1, \ldots, Y_{j-1}, Y_{j+1}, \ldots, Y_p)$ used in the imputation model may themselves be incomplete. We want to avoid specifying too many different imputation models, for example, a model for Y_1 where we use Y_2, Y_3 and Y_4 when they are observed, another model for Y_1 using only Y_2 and Y_3 when Y_4 is also missing, etc. **Multiple imputation using chained equations (MICE)** is a way to overcome this problem.

MICE works by initially filling in values for all the missing data using sampling with replacement from the observed values. This gives us an initial complete data set, which we will call $\tilde{Y}^{(0)}$ (the tilde is to indicate that this is just an intermediate dataset, and that we won't use this in our analysis). We then iterate through the following steps creating a sequence of datasets $\tilde{Y}^{(1)}, \tilde{Y}^{(2)}, \ldots, \tilde{Y}^{(T)}$ as follows:

1. missing values in Y_1 are replaced by $\tilde{Y}_1^{(t)}$ where
$$\tilde{Y}_1^{(t)} \sim f(Y_1 | Y_1^{obs}, \tilde{Y}_2^{(t-1)}, \ldots, \tilde{Y}_p^{(t-1)})$$
2. missing values in Y_2 are replaced by $\tilde{Y}_2^{(t)}$ where
$$\tilde{Y}_2^{(t)} \sim f(Y_2 | \tilde{Y}_1^{(t)}, Y_2^{obs}, \tilde{Y}_3^{(t-1)}, \ldots, \tilde{Y}_p^{(t-1)})$$
3. $\vdots$

We iterate around these steps for a small number of cycles (typically 10-20 iterations) until the algorithm has converged[5]. We take this final dataset, $\tilde{Y}^{(T)}$ say, and set this to be our first imputed dataset $Y^{(1)}$. We then repeat the entire procedure to obtain multiple imputed datasets $Y^{(1)}, \ldots, Y^{(m)}$.

We can recreate something similar to that the procedure used by the `mice` command as follows. Here, `data.mis` contains missing values in `X1` and `X2`, and we will use linear regression to fill in the missing values. The code below creates one imputed dataset by doing 10 cycles around steps 1. to 3. above.

[5] Convergence in the sense of a Markov chain - it is a stochastic algorithm so values will keep changing

```
data.mis
M <- is.na(data.mis) # missing data pattern.
M <- as.data.frame(M)
# Begin by filling in missing values by simulating at random from the data.
data.impute <- data.mis
data.impute$X1[M$X1] <- sample(data.mis$X1[!M$X1], size = sum(M$X1), replace=T)
data.impute$X2[M$X2] <- sample(data.mis$X2[!M$X2], size = sum(M$X2), replace=T)

# Check there are no imcomplete cases left
sum(ici(data.impute))

# We now cycle around prediction steps
for(i in 1:10){
  # first fit a model to predict X1, using the data only in places where we observed X1.
  fit <- lm(X1~Y+X2, data.impute, subset = !M$X1)
  # Then impute using this regression model plus the added error term
  data.impute$X1[M$X1] <- predict(fit, newdata=data.frame(Y=data.impute$Y[M$X1],
             X2 = data.impute$X2[M$X1]))+
             rnorm(sum(M$X1), 0, sd=summary(fit)$sigma)

  # Now fit a model to predict X2, using the data only in places where we observed X2.
  fit <- lm(X2~Y+X1, data.impute, subset = !M$X2)
  data.impute$X2[M$X2] <- predict(fit, newdata=data.frame(Y=data.impute1$Y[M$X2],
            X1 = data.impute1$X1[M$X2]))+
            rnorm(sum(M$X2), 0, sd=summary(fit)$sigma)
}
```

As you can see, using the `mice` command saves us a lot of coding, even in this very simple example.

In practice, simulating from $f(Y_1|Y_{-1}^{(t)}, Y_1^{obs})$ is hard. In the example above, we cheated, and used improper imputation. In other words, to predict missing values of `X1`, we fitted the regression model

$$\texttt{X1} = \beta_0 + \beta_1 \texttt{Y} + \beta_2 \texttt{X2} + \epsilon$$

and then imputed values of `X1` from $f(\texttt{X1}|\tilde{\texttt{Y}}^{(m)}, \widetilde{\texttt{X2}}^{(m)}, \texttt{X1}^{obs}, \hat{\beta})$, i.e., fixing β at its estimated value, $\hat{\beta}$, and ignoring the uncertainty this introduces.

Although approximate approaches like this that were outlined in the previous section (improper imputation or the Bayesian approach based on the complete cases) can work well, we have the opportunity to do something better using the `mice` package. Given that we go through several iterations to produce each imputed dataset, we can also sample multiple different β values. So to sample from $f(Y_{mis}|Y_{obs})$, we can use a Gibbs sampling

algorithm (see MAS6004 for details) and sample from $f(\beta, Y_{mis} \mid Y_{obs})$ by sampling in turn from each of the full conditionals:

$$\beta^{(t)} \sim f(\beta \mid \tilde{Y}_{mis}^{(t-1)}, Y_{obs})$$
$$\tilde{Y}_{mis}^{(t)} \sim f(Y_{mis} \mid Y_{obs}, \beta^{(t)})$$

The first step is now simple as $(Y_{mis}^{(t)}, Y_{obs})$ form a complete dataset, for which we assume the Bayesian approach is easy to apply. We can again initialise the algorithm by setting $\tilde{Y}^{(0)}$ using random sampling from the observed values. When we put this all together, we get an algorithm of the form

1. Sample $\beta_1^{(t)} \sim f(\beta_1 \mid Y_1^{obs}, \tilde{Y}_2^{(t-1)}, \ldots, \tilde{Y}_p^{(t-1)})$ where β_1 is the regression parameter used to predict Y_1 from Y_{-1}.
2. Sample new values of the missing Y_1 values, denoted $\tilde{Y}_1^{(t)}$:

$$\tilde{Y}_1^{(t)} \sim f(Y_1 | Y_1^{obs}, \tilde{Y}_2^{(t-1)}, \ldots, \tilde{Y}_p^{(t-1)}, \beta_1^{(t)})$$

3. Sample $\beta_2^{(t)} \sim f(\beta_2 \mid \tilde{Y}_1^{(t)}, Y_2^{obs}, \tilde{Y}_3^{(t-1)}, \ldots, \tilde{Y}_p^{(t-1)})$ where β_2 is the regression parameter used to predict Y_2 from Y_{-2}.
4. Sample new values of the missing Y_2 values, denoted $\tilde{Y}_2^{(t)}$:

$$\tilde{Y}_2^{(t)} \sim f(Y_2 | \tilde{Y}_1^{(t)}, Y_2^{obs}, \tilde{Y}_3^{(t-1)}, \ldots, \tilde{Y}_p^{(t-1)}, \beta_2^{(t)})$$

5. Sample $\beta_3^{(t)}$...
6. etc
7. Repeat steps above until convergence

Note that this approach solves all of the problems we listed for single imputation methods:

1. It is stochastic, using the both the structured and random parts of the regression model
2. It deals with parametric uncertainty by sampling the unknown parameter values from the range of plausible values
3. It uses all the data, not just the complete cases

Coding all this up ourselves for each problem would be extremely time consuming. Thankfully, the `mice` package has done all this for us for most of the imputation approaches we would want to use. It is extremely flexible and gives us great control over how we do the imputation. For example, in some problems, it can be

advantageous to change the order in which we cycle through the columns of Y depending on the pattern of missingness. It also allows us to control what covariates are used to fill in the missing values for each covariate, as well as giving us a wide range of choices for the imputation model we use (see the next section). For details, and for a fuller description of the MICE approach (and details of how to use the `mice` R package) see van Buuren and Groothuis-Oudshoorn 2011, which is available on MOLE.

7.2.2 Building models for imputation

The type of regression we use to impute the missing data depends on what type of data it is. For continuous data, the most sensible approach is often to use linear regression. For example,

$$Y_1 = \beta_1 + \beta_2 Y_2 + \ldots \beta_p Y_p + \epsilon$$

If Y_1 is a binary variable, then instead of using linear regression to impute its value, we might use a logistic regression model, for example, assuming

$$\text{logit}\, P(Y_1 = 1 | Y_{-1}^{(t-1)}, \beta) = \beta_1 + \beta_2 Y_2 + \ldots \beta_p Y_p$$

For an unordered categorical variable, we could use a multinomial regression model etc. If the variables are constrained in some way (to be positive for example), then we should take that into account (e.g. by taking logs). In the notes, I have described only Bayesian regression type models for doing the imputation, but we are free to use any type of method we choose.

The advantage of the MICE approach is that we can use different types of imputation model for each variable. Thankfully, these approaches are all automated in the `mice` R package, making it beautifully simple to use these approaches. The `mice` package also allows us to control which variables to use in the imputation model. The general rule is to use as many predictors as we can (i.e., using all the available information), as this makes the MAR assumption more plausible. However, if the data contains 100s of covariates this can be impossible, and so in this case we would try to pick a subset of may 15-25 covariates to use. Van Buuren and Groothuis-Oudshoorn 2011 contains advice on how to choose these variables.

7.2.3 Brief description of using mice

Once we have created the m imputed datasets, it is time to perform the statistical analysis (see Figure 7.1). We use the

`with` command to apply the statistical method to each imputed dataset in turn.

This creates m different parameter estimates, one for each imputed dataset. To combine them, we can use the `pool` command, which applies Equations (7.2) and (7.3). This works for a wide range of statistical methods in R, including on output from the `lm`, `glm`, `Arima`, and `aov` amongst others. The `pool` command will also work on output from the `lmer` functions. Note, however, that this is simply combining the coefficient estimates and the variance-covariance estimates from `lmer`. Imputing multilevel data is challenging, and how to do so is still an open and active research area. You cannot, for example, in general simply use the `mice` command to impute multilevel data.

This module doesn't cover these intracies, or how to do hypothesis tests with multiple imputation, for example for model selection. There is an approach implemented in `mice` that will calculate p-values, but again, care is needed. The details can again be found in Van Buuren and Groothuis-Oudshoorn 2011.

7.3 Discussion

The EM algorithm required us to specify a complete joint multivariate distribution for Y. We then performed inference using a likelihood based approach. This is, in some sense, the best approach we could use if we can specify a sensible multivariate distribution for Y (and are happy to use maximum likelihood). We could use the same joint modelling approach with multiple imputation, but it would require us to simulate from the conditional distributions, which can be hard (as we saw for the multivariate normal example - the simplest model of interest!), and it wouldn't be as statistically efficient as using the EM algorithm.

The advantage of MI, is that we can create the multiple imputed datasets without consideration of the precise model which will subsequently be used to analyse the data (i.e., it doesn't matter whether eventually we plan to fit a mixed effects model, or a fixed effects model, or a time-series model, etc, we can first do the multiple imputation without considering what it will be used for). Somewhat surprisingly, empirical evidence suggests that this can work well in a variety of situations. Fully conditional specifications, where we specify a model for the joint distribution of $Y_1|Y_2, \ldots, Y_p$, a model for $Y_2|Y_1, Y_3, \ldots, Y_p$ etc, can work well even when these individual conditional models do not imply a sensible joint model. As van Buuren and Groothuis-Oudshoorn say, the '*MICE algorithm possesses a touch of magic*'.