Advancing in R

Module 8: mixed models

Introduction

In this practical exercise, we will introduce the analysis of mixed models.

We will assume that you are familiar with the general syntax of coding in R, that you are adept at making basic plots, and that you have reviewed material on linear models with continuous and factorial predictors. If you need to, review this material before the session starts.

Learning outcomes

Know how to specify random effects in mixed models using lme4

Know how to interpret variance components

Know how to simplify mixed models, and thereby conduct hypothesis tests on fixed effects within them

Know to report parameters from models fit with REML, and compare models fit with ML

Know how to specify random effects structures with crossed fixed effects (supplementary exercises)

Know how to adopt predict() to illustrate mixed models

Libraries

In addition to the basic libraries automatically loaded with R, you may wish to use the following additional packages. If necessary, make sure you either have internet access on your machine of that you install these packages prior to the practical.

{arm} {lme4} {Matrix} {Rcpp} {psych}

Some useful commands

Below is a list of some (but not all) of the useful commands you may need to run at some stage of this practical exercise.

AICc()

expand.grid()

levels()

lm()

lmer()

log()

predict()

xtabs()

The data set for our first exercise features a single key response variable, the number of grouse that are shot at a particular location, in relation to two predictor variables: the number of grouse counted in an earlier survey of the site and the number of previous shooting events on the site in question. Grouse were counted in specific areas and shot in the same area about one month later. The authors of this study expected that the survey counts would predict the number of grouse shot, but that the number of previous shootings would also be important. Because each location (a drive, which is one part of a moor) could be visited more than once, and because adjacent drives within moors could be sampled, the data may be neither spatially nor temporally independent. More details can be found in Bunnefeld et al. (2009).

Open a new project and start a new script. Load the data (grouse_shooting_mod.csv, available in the course materials) into a descriptively named object. Inspect the data structure, and ensure that the vectors are being correctly interpreted. Conduct the usual quality control steps to search for data entry mistakes.

The data are currently not yet ready for analysis, because the number of birds shot is not yet standardized by the size of the sampling location. Generate a new variable within your data frame that will serve as the response by dividing the number of birds shot (nr_shot) by the drive area (km2). You will want to log-transform both this newly generated variable and the predictor for the number of birds estimated in the earlier survey (totalcount). We will use a linear model with natural log transformations (rather than a generalized linear model) because we would much prefer to have a slightly less interpretable but analytically straightforward model than one that is arguably more technically correct, but more difficult to evaluate.

Having generated the required new response and predictor, examine the distributions of both variables and the remaining predictor (prev), and make annotations if you have any special concerns about these.

Your next step will be to plot the effects of prev and survey count so that you can predict the coefficients of the model. Make sure that your plot allows you to visualize the effects of both predictors within the same panel so that you can tell if there is likely to be a significant interaction. Write down your predictions. (For simplicity, let's model the number of previous shootings (prev) as a continuous predictor, but note that Bunnefeld et al. (2009) it is modeled as a factor. Can you think of reasons supporting each approach?)

Now it is time to start modeling. Rather than get straight to the mixed modeling, let's first build a simpler model with a fixed effects structure only, so that we can get a sense for the model notwithstanding the possible problems with pseudoreplication. This is always a good idea for approaching complex problems: first try to work on the simplest version so that you can better recognize what might be going right or wrong, before trying to execute a really complicated procedure all at once.

Build a fixed effects only linear model using the lm() function, which the predictors are log-transformed count and prev, and the response is log transformed number of birds shot per km^2. Assess the model diagnostics, and conduct simplification to obtain a minimal adequate model. Do "prev" shooting events and log-transformed survey counts predict log (shooting success)? Does the effect of one variable depend on the other? How do the coefficients compare to your guesses

© Luc F. Bussière & Tom M. Houslay, November 2015

based on the exploratory plot you made earlier? Why is this model insufficient for drawing conclusions about the value of these predictors for estimating grouse shot?

One suggestion for dealing with the pseudoreplication in this model is to include a fixed factor that fits the effects of both drive and moor (here treated as fixed categorical predictors) on the response. Try this now. How many coefficients does the summary indicate there should be? How many values are actually generated? Why are many of the estimates simply NA? To help you answer this question, generate a table that counts how many times each combination of moor, drive, and prev shooting instances have been sampled using the following code:

```
xtabs(~moor+drive,data=GROUSE)

xtabs(~moor+drive+prev,data=GROUSE)
```

You will notice how poorly the combinations have been sampled; this is not an indictment of the authors, but rather an indication of how difficult it is to get balanced and large data sets. In order to circumvent the fact that fitting all the moor and drive identity would suck up more df than we have to spare, we can make both drive and moor random effects. This saves the model having to use df to assess them, and it conveniently also deals with the possibility that measures within a single area are more likely to be similar (which suggests probably spatial and temporal pseudoreplication).

If necessary, install the library {lme4} and load it into the RStudio memory. The syntax for calling a mixed effects model is as follows:

```
g1.mixed<-lmer(shot~prev*count+(1|moor)+(1|drive),data=GROUSE)
```

This syntax features both familiar and new components. On the left, we can easily recognize the structure of the formula indicating the fixed effects of the model. However, there are two components in brackets that may look strange: **+(1|moor)+(1|drive)** is the random effects structure for this model. The 1 prior to the vertical line tells R that we only want to allow intercepts to vary with each of the random effects (rather than allowing the fixed effects to vary as well – we'll try that more complex maneuver in the supplementary exercises). This means that rather than using df to estimate the intercept and slope deviations for each level in these two vectors, we want R to soak up the variance associated with both moor and drive, while still focusing on estimating the coefficients for the fixed effects: log(count) and prev.

You may want to call a plot of this model, but by default it will only generate a single diagnostic plot of fitted on residual values. There are a number of additional options for investigating other diagnostics (e.g., sjp.lmer(), which will require the {arm} and {sjPlot} libraries, and will illustrate variation in the random effects levels), but because we have already inspected the diagnostics for the fixed version, we can be reasonably assured that our model is unlikely to be atrocious.

Having satisfied yourself that the diagnostics don't flag any serious issues, examine the model summary. Notice that this includes the usual reminder of model structure, scaled residuals, a new section describing the variance components, and then the table of fixed effects, which corresponds to the coefficients from fixed effects models. Finally, the bottom of the output provides the correlation matrix for fixed effects. This information is rarely reported as it is a rather technical

© Luc F. Bussière & Tom M. Houslay, November 2015

aspect of the computations used to generate mixed models, although in some cases it specification or derivation are of specific interest.

Examine the random effects component of the output. Notice how this section gives us more information on the number of observations as well as how many moors and drives there are in our dataset. Now compare the magnitude of the estimated variance components for moor, drive, and residual error. What fraction of the total variance (the sum of these three components) is explained by moor? What fraction is explained by drive?

Now examine the fixed effects. How do the coefficients compare to those for the analogous model that did not include any random effects? How do they compare to your own guesses of coefficients (based on your exploratory plot)? Then notice that the table of fixed effects provides no p-values. There is a good reason for this: in order to assess p-values, we need to know the degrees of freedom available for a given term. However, by fitting random effects we are acknowledging that our observations are not necessarily independent, which means that we have fewer df than our observations suggest. How many fewer, exactly, is unclear, and there is ongoing debate about how best to conduct hypothesis testing for coefficients from mixed models. Luckily for us, there is a familiar approach that we can adopt to derive hypothesis tests: we can simply exclude a focal fixed component of the model, and then conduct a likelihood ratio test. Because the LR-test involves a known number of coefficients, we know the df for the LR test with much more confidence than we know the df for any individual t-value in the table of coefficients. And while using LR tests is by no means universally accepted, many ecologists and evolutionary biologists accept it as a moderately objective way to select among mixed models.

Use the update command to remove the interaction term from your mixed model, and then the anova() function to compare the simplified model to the original one. Notice that R will let you know that it is refitting your mixed models prior to comparing them. By default, when estimating your coefficients lme4 uses restricted maximum likelihood (REML), because this procedure is thought to generally provide more robust parameter estimates than maximum likelihood (ML). However, we cannot compare models fit with REML, so lme4 conveniently refits using ML when we ask for an anova comparing two models (until recently, we had to remember to do the refitting by changing the method argument in the model call, but the most recent updates to lme4 have made our lives easier). Does the likelihood ratio test support the simpler or the more complex model? Does this agree with the model selection procedure you conducted on the fixed effects model without any random effects earlier? Continue trying to simplify the model until you know that you have the minimal adequate model. While you do this, you will be conveniently generating p-values for all of the fixed effects in your model, but these p-values will come from the LR tests instead of from the table of coefficients. When you report coefficients in your results, you can cite the Chi-squared test statistic and p-value associated with this likelihood ratio test along with the coefficient and standard error.

Alternatively (or in addition), you may wish to compute the AIC's for your different models. In fact, because our sample size is small, we will probably want to calculate AICc's, which are corrected for the small sample size. One implementation of the AICc calculation is in the MuMIn library, which you can execute by installing MuMIn (if necessary), loading it, and running the function

© Luc F. Bussière & Tom M. Houslay, November 2015

AICc() on the models you wish to compare. Does this procedure accord with model simplification using LR tests?

Once you are satisfied with the model and your interpretation of it, generate a publication quality figure that illustrates the data and the fitted effects using the fixed effects coefficients from the summary of your minimum adequate model. Make sure you have adequately annotated your script, and then save it along with your figure.

~ End of Practical ~

© Luc F. Bussière & Tom M. Houslay, November 2015