Advancing deterministic algorithms for mixed models in R

Steven C. Walker, Ben Bolker (McMaster University)

Aug 11-18, 2013

1 Overview of the Field

> "Science is what we understand well enough to explain to a computer. Art is everything else we do." – Donald Knuth

Mixed effects modelling provides a flexible and well-defined approach to the statistical analysis of data sets with correlated replicate measurements. A wide variety of computational tools are available for fitting mixed models. Still, there are many mixed model problems for which statistical software is unavailable, too expensive, or too difficult for the vast majority of potential users. Put differently, although mixed modelling is a very old, useful, and well-studied topic—going back to the founders of statistics—it continues to present subtle challenges that are difficult "...to explain to a computer". Our central goal with this workshop is to expand the range and quality of computational tools available for analyzing data with mixed models, focusing on the `lme4` package (and its relatives) in `R`. We focus on `lme4` because it is (1) widely used; (2) being actively developed; (3) relatively efficient with big data; and (4) flexible.

The `lme4` package is based on the generalized linear mixed model,

$$\underbrace{y_i}_{\text{response}} \sim \underbrace{\text{Distr}}_{\substack{\text{conditional}\\ \text{distribution}}} \left(\underbrace{g^{-1}(\eta_i)}_{\substack{\text{inverse}\\ \text{link}\\ \text{function}}}, \underbrace{\phi}_{\substack{\text{scale}\\ \text{parameter}}}\right)$$

$$\underbrace{\eta}_{\substack{\text{linear}\\ \text{predictor}}} = \underbrace{X\beta}_{\substack{\text{fixed}\\ \text{effects}}} + \underbrace{Zb}_{\substack{\text{random}\\ \text{effects}}}$$

$$\underbrace{b}_{\substack{\text{random}\\ \text{coefficients}}} \sim \underbrace{\text{MVN}}_{\substack{\text{multivariate}\\ \text{normal}}} \left(0, \underbrace{\Sigma(\theta)}_{\substack{\text{variance-}\\ \text{covariance}\\ \text{matrix}}}\right)$$

where y_i is the ith observed response, η_i the linear predictor of y_i, $g^{-1}(\eta_i)$ the mean of y_i, ϕ a possibly unknown scale parameter, η is an n-vector containing the η_i, X an n-by-p model matrix for fixed effects, Z an n-by-q model matrix for random-effects, β and b are p- and q-vectors of fixed- and random-effects coefficients, θ is a vector of covariance parameters on which the variance covariance matrix, $\Sigma(\theta)$, of b depends. The `lme4` package estimates β and θ by maximum likelihood (or restricted maximum likelihood). The likelihood function, $\mathcal{L}(y_i, \theta, \beta)$, for the ith response involves evaluating an integral over the random effects coefficients,

$$\mathcal{L}(y_i, \theta, \beta) = \int \mathcal{L}(y_i|\theta, b') \times \mathcal{L}(b'|\Sigma(\theta))db'$$

An lme4 model is specified through an extension of R's formula notation,

$$\begin{aligned}\texttt{response} \sim\ & \texttt{predictor}_1 + \texttt{predictor}_2 + ... + \\ & (\texttt{predictor}_1 + \texttt{predictor}_2 + ... \mid \texttt{block}_1) + \\ & (\texttt{predictor}_1 + \texttt{predictor}_2 + ... \mid \texttt{block}_2) \ + ...\end{aligned}$$

Here the response variable is modelled using predictor variables and blocking variables. The predictor variables outside of bracketed terms go in the fixed effects design matrix, X, and those inside bracketed terms in the random effects design matrix, Z. In general, a mixed-model formula incorporates one or more random-effects terms of the form (r|f) where r is an R-language linear model formula and f is a grouping factor. This method of input allows users to specify a model with a particular structure for Z and $\Sigma(\theta)$, which is useful for modelling a wide variety of data that come in blocks (e.g. longitudinal data).

The lme4 package has several advantages including,

- It can fit models with an exponential family as the conditional distribution for the response variable, and therefore can handle both linear and generalized linear mixed models (LMMs, GLMMs).
- It allows arbitrarily many nested and crossed random effects.
- Uses efficient sparse matrix algebra algorithms from the Eigen package, which are very efficient.

These advantages help make lme4 one of the most used R packages. For example, lme4 was the 44th most downloaded of 4866 R packages downloaded between 9 June and 4 August 2013 from probability.ca. Furthermore, 433 other R packages depend on lme4. According to Google Scholar, lme4 has been cited 3041 times in published academic work.

2 Open Problems Addressed at BIRS

2.1 User Interfaces

Specifying uncorrelated random effect coefficients is tedious Consider a case in which the effect of a categorical variable, x, takes on different random values in different levels of a grouping variable, g. Specifying a model that accounts for both (1) the variances of the random effects of each level of x among the levels of g and (2) the correlations between these effects is straightforward in lme4. In particular, the specification of this random effects structure is (x|g). However, we often want to assume that the random effects are uncorrelated. To do this, we need to specify the random structure for each level of x, which is done using an expression such as $(\texttt{x}_1|\texttt{g}) + (\texttt{x}_2|\texttt{g}) + ...$ When x has many factors this can become very tedius. While at BIRS, Fabian Scheipl developed the expression-parsing machinery in lme4 to allow more succinct notation for uncorrelated random effects. In particular, $(\texttt{x}||\texttt{g})$ can be used in place of $(\texttt{x}_1|\texttt{g}) + (\texttt{x}_2|\texttt{g}) + ...$

Default printouts of model fit information can be too verbose It can be difficult to construct an intellegent algorithm for determining default printout behaviour. For example, some users will want more information and others less. Still, it is important to develop defaults that satisfy as many users' requirements as possible. At BIRS, Martin Maechler made substantial progress in this regard.

Parametric bootstrap: `pbkrtest` One of the important advances of the lme4 package was the ability to use non-Gaussian distributions for response variables, resulting in generalized linear mixed models (GLMMs). A challenge in GLMMs is that there is no general analytic theory for computing confidence intervals and distributions under null hypotheses. One potential solution in theory is to use the parametric bootstrap, which is a Monte-Carlo simulation method. However, there is a general lack of software for conducting such bootstrap methods, leaving users to write their own problem-specific code. While at BIRS, Soren Hojsgaard extended his pbkrtest package so that the parametric bootstrap could be used generally for any GLMM that can be fitted by lme4.

linest After estimating a linear mixed model, many users prefer to categorical variables using the mean effects for each level (so called least-squares means). lme4 requires that users estimate these means manually or with custom code. While at BIRS, Soren Hojsgaard extended his doBy package to calculate least-squares means from lme4-fitted models.

2.2 Developer Interfaces

The computational efficiency and flexbility of lme4 makes it a powerful tool. Still, in statistical software development, there is often a trade-off between efficiency and code readability. Because lme4 uses several efficient but advanced computational tools (e.g. the C++ linear algebra library, Eigen; the R paradigm of reference classes), the internal code can be difficult to read and therefore difficult to extend. In order to help R developers make use of the power of lme4 in their statistical software, we have provided several novel ways to hook into various stages of the mixed model fitting process.

Plain lme4: lme4pureR While at BIRS, Doug Bates and Steve Walker developed the lme4pureR package, which is written entirely in R, without any compiled code. We hope the simplicity of lme4pure will make it easier for developers to prototype and test new mixed model formulations.

General formulation for estimating structured random effects covariance matrices There is a fairly hard trade-off faced by mixed-model analysts in R when choosing between the older and slower nlme package and the newer and faster lme4 package. In particular, nlme can fit a wide variety of correlation structures (e.g. auto-regression; geo-statistical models) but cannot fit GLMMs, where as lme4 can fit GLMMs but is relatively inflexible in terms of fitting structured covariance matrices. A major advance at our BIRS focused research group was the development of a new version of lme4 that allows much greater flexibility in the specification of covariance structures (Fabian Scheipl, Doug Bates, and Ben Bolker).

Incorporating Bayesian boundary avoiding priors: blme A very common problem with fitting mixed models to relatively small data sets is that estimates of covariance parameters can be on the boundary of the parameter space. For example, a random effect may be estimated to have zero variance, even though we know that there must be some among-block variation. In order to include this prior information that variance parameters should not be on the boundary, Vince Dorie extended the definition of the objective function in lme4. This advance will help developers build more boundary-avoiding tools for lme4 and related packages.

Modularization of the major steps of lme4 model fitting When lme4 fits a model, it goes through four largely independent steps: formula-parsing, creation of the objective function, optimizing the objective, and packaging up the results for user output. Previously, it was not possible for a developer to modify the results of any one step, before passing those results to the next step. Ben Bolker and Steve Walker have modularized the lme4 fitting functions into four sub-functions that represent these steps. This functionality has already been incorporated in the gamm4 package.

2.3 Theory

PIRLS lme4 fits generalized linear mixed models using an algorithm developed by Doug Bates called penalized iteratively reweighted least squares (PIRLS), which is a modification of the iteratively reweighted least squares algorithm for fitting generalized linear models. Before coming to BIRS, the theoretical justification for PIRLS was still preliminary. At BIRS, Rune Haubo Bojesen Christensen, Doug Bates, Martin Maechler, and Steve Walker solidified the derivation of the algorithm, identifying more clearly the theoretical conditions for convergence.

Gauss-Hermite quadrature For models with only a single scalar random effect, lme4 can approximate the integral in the expression for the likelihood function above using adaptive Gauss-Hermite quadrature. For all other models, the less accurate Laplace approximation is used. While at BIRS, Rune Haubo B Christensen developed theory for extending the applicability of quadrature methods to certain nested models with vector-valued random effects. He also presented evidence that it might be more efficient

to use non-adaptive quadrature (i.e. quadrature points centered around zero instead of the mode). By centering around zero, you do not need to waste computations finding the mode, and the nature of certain link functions (e.g. logit) essentially ensures that the mode will never be too far from zero. Doug Bates suggested a hybrid approach that takes a few adaptive steps towards the mode.