Multilevel Models in R: Present & Future

Douglas M. Bates

U of Wisconsin - Madison, U.S.A.

useR!2004 – Vienna, Austria

Overview

- What are multilevel models?
- Linear mixed models
- Symbolic and numeric phases
- Generalized and nonlinear mixed models
- Examples

What are multilevel models?

- The term was coined by Harvey Goldstein and colleagues working at the Institute of Education in London and applied to models for data that are grouped at multiple "levels", e.g. student within class within school within district. Goldstein, Rasbash, and others developed computer programs ML3 (three levels of random variation), MLn (arbitrary number of levels), and MLWin (user-friendly version for Windows) to fit models to such data.
- Similar ideas were developed by Raudenbush and Bryk (Michigan and Chicago) under the name "hierarchical linear models" and incorporated in a program HLM.
- Both programs were extended to handle binary responses (i.e. Yes/No, Present/Absent, ...) or responses that represent counts.

What are multilevel models? (cont'd)

- To statisticians multilevel models are a particular type of mixed-effects model. That is, they incorporate both
  fixed effects: parameters that apply to the entire population or well-defined subgroups of the population.
  random effects: parameters that apply to specific experimental units, which represent a sample from the population of interest.
- The model is written in terms of the fixed-effects parameters and the variances and covariances of the random effects.

Linear mixed models

We will write the linear mixed model as

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{Z}\boldsymbol{b} + \boldsymbol{\epsilon} \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \sigma^2\boldsymbol{I}), \boldsymbol{b} \sim \mathcal{N}(\boldsymbol{0}, \sigma^2\boldsymbol{\Omega}^{-1}), \boldsymbol{\epsilon} \perp \boldsymbol{b}$$

where

- The response is $\boldsymbol{y}$ (n-dimensional).
- The $n \times p$ model matrix $\boldsymbol{X}$ and the $n \times q$ $\boldsymbol{Z}$ are associated with the fixed effects $\boldsymbol{\beta}$ and the random effects $\boldsymbol{b}$.
- The "per-observation" noise $\boldsymbol{\epsilon}$ is spherical Gaussian.
- The relative precision of the random effects is $\boldsymbol{\Omega}$.

Structure in the random effects

- The random effects $\boldsymbol{b}$ are associated with one or more grouping factors $\boldsymbol{f}_1, \ldots, \boldsymbol{f}_k$, each of length n.
- The number of distinct values in $\boldsymbol{f}_i$ is m_i. Typically at least one of the m_i is on the order of n.
- Each grouping factor is associated with an $n \times q_i$ model matrix $\boldsymbol{Z}_i$. The q_i are often very small. For a variance components model $q_1 = \cdots = q_k = 1$.
- The random effects vector $\boldsymbol{b}$ is divided into k outer groups, corresponding to the grouping factors. Each of these is subsequently divided into m_i inner groups of length q_i, corresponding to levels of that grouping factor.
- We assume that the outer groups are independent ($\boldsymbol{\Omega}$ is block diagonal in k blocks of size $m_i q_i$) and the inner groups are i.i.d. (each block of $\boldsymbol{\Omega}$ is itself block diagonal consisting of m_i repetitions of a $q_i \times q_i$ matrix $\boldsymbol{\Omega}_i$).
- Each $\boldsymbol{\Omega}_i$ is a symmetric, positive-definite matrix determined by a $q_i(q_i+1)/2$ parameter $\boldsymbol{\theta}_i$. These are collected into $\boldsymbol{\theta}$.

Exam scores in inner London

The exam scores of 4,059 students from 65 schools in inner London are an example used by Goldstein, Rasbash et al. (1993).

```
> str(Exam)
'data.frame':        4059 obs. of  8 variables:
 $ school  : Factor w/ 65 levels "1","2","3","4",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ normexam: num   0.261  0.134 -1.724  0.968  0.544 ...
 $ standlrt: num   0.619  0.206 -1.365  0.206  0.371 ...
 $ gender  : Factor w/ 2 levels "F","M": 1 1 2 1 1 2 2 2 1 2 ...
 $ schgend : Factor w/ 3 levels "mixed","boys",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ schavg  : num  0.166 0.166 0.166 0.166 0.166 ...
 $ vr      : Factor w/ 3 levels "bottom 25%","mi..",..: 2 2 2 2 2 2 2 2 2 2 ...
 $ intake  : Factor w/ 3 levels "bottom 25%","mi..",..: 1 2 3 2 2 1 3 2 2 3 ...
```

Exam score grouping factor

The (sole) grouping factor in the *Exam* data is *school*.

```
> summary(Exam[, "school", drop = FALSE])
     school
 14     : 198
 17     : 126
 18     : 120
 49     : 113
 8      : 102
 15     :  91
 (Other):3309
```

Models fit to these data will have $n = 4059$, $k = 1$ and $m_1 = 65$.

Scottish secondary school scores

Scores attained by 3435 Scottish secondary school students on a standardized test taken at age 16. Both the primary school and the secondary school that the student attended have been recorded.

```
> str(ScotsSec)
'data.frame':        3435 obs. of  6 variables:
 $ verbal : num  11 0 -14 -6 -30 -17 -17 -11 -9 -19 ...
 $ attain : num  10 3 2 3 2 2 4 6 4 2 ...
 $ primary: Factor w/ 148 levels "1","2","3","4",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ sex    : Factor w/ 2 levels "M","F": 1 2 1 1 2 2 2 1 1 1 ...
 $ social : num  0 0 0 20 0 0 0 0 0 0 ...
 $ second : Factor w/ 19 levels "1","2","3","4",..: 9 9 9 9 9 9 1 1 9 9 ...
```

ScotsSec grouping factors

```
> summary(ScotsSec[, c("primary", "second")])
    primary         second
 61     :  72   14     : 290
 122    :  68   18     : 257
 32     :  58   12     : 253
 24     :  57   6      : 250
 6      :  55   11     : 234
 1      :  54   17     : 233
 (Other):3071   (Other):1918
```

Models fit to these data have $n = 3435$ and $k = 1$ or $k = 2$. When $k = 2$, $m_1 = 148$ and $m_2 = 19$.

Scores on 1997 A-level Chemistry exam

Scores on the 1997 A-level Chemistry examination in Britain. Students are grouped into schools within local education authorities. Some demographic and pre-test information is also provided.

```
> str(Chem97)
'data.frame':        31022 obs. of  8 variables:
 $ lea      : Factor w/ 131 levels "1","2","3","4",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ school   : Factor w/ 2410 levels "1","2","3","4",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ student  : Factor w/ 31022 levels "1","2","3","4",..: 1 2 3 4 5 6 7 8 9 10 ...
 $ score    : num  4 10 10 10 8 10 6 8 4 10 ...
 $ gender   : Factor w/ 2 levels "M","F": 2 2 2 2 2 2 2 2 2 2 ...
 $ age      : num  3 -3 -4 -2 -1 4 1 4 3 0 ...
 $ gcsescore: num  6.62 7.62 7.25 7.50 6.44 ...
 $ gcsecnt  : num  0.339 1.339 0.964 1.214 0.158 ...
```

Chem97 grouping factors

```
> summary(Chem97[, c("lea", "school", "gender")])
      lea            school       gender
 118    :  969   698    :  188   M:17262
 116    :  931   1408   :  126   F:13760
 119    :  916   431    :  118
 109    :  802   416    :  111
 113    :  791   1215   :   99
 129    :  762   908    :   94
 (Other):25851   (Other):30286
```

Models fit to these data have $n = 31022$ and $k = 1$ or $k = 2$. When $k = 2$, $m_1 = 2410$ and $m_2 = 131$.

Dallas TLI scores

The 'dallas' data frame has 369243 rows and 7 columns. The data are the results on the mathematics part of the Texas Assessment of Academic Skills (TAAS) for students in grades 3 to 8 in the Dallas Independent School District during the years 1994 to 2000. Because all the results for any student who took a test in the Dallas ISD are included, some of the test results are from outside the Dallas ISD.

```
> str(dd)
'data.frame':        369243 obs. of  7 variables:
 $ ids   : Factor w/ 134712 levels "6","16","22",..: 1 2 3 4 5 6 6 7 7 8 ...
 $ Sx    : Factor w/ 2 levels "F","M": 2 2 2 1 1 1 1 2 2 2 ...
 $ Eth   : Factor w/ 5 levels "B","H","M","O",..: 2 1 2 2 1 2 2 2 2 1 ...
 $ Year  : int  1997 1998 2000 2000 1995 1994 1995 1994 1995 1994 ...
 $ Gr    : int  4 7 7 7 4 7 8 3 4 3 ...
 $ Campus: Factor w/ 887 levels "1907041","1907110",..: 269 136 147 133 245 140 14
 $ tli   : int  63 66 54 75 56 64 62 74 88 74 ...
```

Dallas grouping factors

```
> summary(dd[, c("ids", "Campus")])
      ids              Campus
 1075648:    12   57905049: 10499
 2306440:    11   57905051:  6177
 2399735:    10   57905043:  5784
 2588394:    10   57905042:  5581
 3134529:    10   57905065:  5342
 686265 :     9   57905052:  5151
 (Other):369181   (Other) :330709
```

Models fit to these data have $n = 369243$ and $k = 1$ or $k = 2$. When $k = 2$, $m_1 = 134712$ and $m_2 = 887$.

A penalized least squares problem

- We seek the maximum likelihood (ML) or the restricted (or residual) maximum likelihood (REML) estimates of the parameters $\boldsymbol{\beta}$, σ^2, and $\boldsymbol{\theta}$.
- The conditional estimates $\widehat{\boldsymbol{\beta}}(\boldsymbol{\theta})$ and $\widehat{\sigma^2}(\boldsymbol{\theta})$ and the conditional modes $\widehat{\boldsymbol{b}}(\boldsymbol{\theta})$ can be determined by solving a penalized least squares problem, say by using the Cholesky decomposition.

$$\begin{bmatrix} \boldsymbol{Z}^\mathsf{T}\boldsymbol{Z}+\boldsymbol{\Omega} & \boldsymbol{Z}^\mathsf{T}\boldsymbol{X} & \boldsymbol{Z}^\mathsf{T}\boldsymbol{y} \\ \boldsymbol{X}^\mathsf{T}\boldsymbol{Z} & \boldsymbol{X}^\mathsf{T}\boldsymbol{X} & \boldsymbol{X}^\mathsf{T}\boldsymbol{y} \\ \boldsymbol{y}^\mathsf{T}\boldsymbol{Z} & \boldsymbol{y}^\mathsf{T}\boldsymbol{X} & \boldsymbol{y}^\mathsf{T}\boldsymbol{y} \end{bmatrix} = \boldsymbol{R}^\mathsf{T}\boldsymbol{R}, \boldsymbol{R} = \begin{bmatrix} \boldsymbol{R}_{ZZ} & \boldsymbol{R}_{ZX} & \boldsymbol{r}_{Zy} \\ \boldsymbol{0} & \boldsymbol{R}_{XX} & \boldsymbol{r}_{Xy} \\ \boldsymbol{0} & \boldsymbol{0} & r_{yy} \end{bmatrix}$$

where $\boldsymbol{R}_{ZZ}$ and $\boldsymbol{R}_{XX}$ are upper triangular and non-singular. Then

$$\begin{aligned} \boldsymbol{R}_{XX}\widehat{\boldsymbol{\beta}}(\boldsymbol{\theta}) &= \boldsymbol{r}_{Xy} \\ \boldsymbol{R}_{ZZ}\widehat{\boldsymbol{b}}(\boldsymbol{\theta}) &= \boldsymbol{r}_{Zy} - \boldsymbol{R}_{ZX}\widehat{\boldsymbol{\beta}} \end{aligned}$$

Estimation criteria

- The profiled estimation criteria, expressed on the deviance (negative twice the log-likelihood) scale are

$$-2\tilde{\ell}(\boldsymbol{\theta}) = \log\left(\frac{\left|\boldsymbol{Z}^\mathsf{T}\boldsymbol{Z}+\boldsymbol{\Omega}\right|}{|\boldsymbol{\Omega}|}\right) + n\left[1+\log\left(\frac{2\pi r_{yy}^2}{n}\right)\right]$$

$$-2\tilde{\ell_R}(\boldsymbol{\theta}) = \log\left(\frac{\left|\boldsymbol{Z}^\mathsf{T}\boldsymbol{Z}+\boldsymbol{\Omega}\right||\boldsymbol{R}_{XX}|^2}{|\boldsymbol{\Omega}|}\right) + (n-p)\left[1+\log\left(\frac{2\pi r_{yy}^2}{n-p}\right)\right]$$

- $\log|\boldsymbol{\Omega}| = \sum_{i=1}^{k} m_i \log|\boldsymbol{\Omega}_i|$ and the $\log|\boldsymbol{\Omega}_i|$ are easy to evaluate because q_i is small.
- $\left|\boldsymbol{Z}^\mathsf{T}\boldsymbol{Z}+\boldsymbol{\Omega}\right| = |\boldsymbol{R}_{ZZ}|^2$ is easy to evaluate from the triangular $\boldsymbol{R}_{ZZ}$. In fact we use an alternative form of the Cholesky decomposition as $\boldsymbol{Z}^\mathsf{T}\boldsymbol{Z}+\boldsymbol{\Omega} = \boldsymbol{L}\boldsymbol{D}\boldsymbol{L}^\mathsf{T}$ where $\boldsymbol{L}$ is unit lower triangular and $\boldsymbol{D}$ is diagonal with positive diagonal elements. Then $\log\left|\boldsymbol{Z}^\mathsf{T}\boldsymbol{Z}+\boldsymbol{\Omega}\right| = \sum_{j=1}^{q}\log d_{jj}$.

Obtaining the Cholesky decomposition

- At this point it is a "mere computational problem" to obtain the REML or ML estimates for any linear mixed model. The little detail we need to work out is how to factor $\boldsymbol{Z}^\mathsf{T}\boldsymbol{Z}+\boldsymbol{\Omega}=\boldsymbol{L}\boldsymbol{D}\boldsymbol{L}^\mathsf{T}$.
- Although $\boldsymbol{Z}^\mathsf{T}\boldsymbol{Z}+\boldsymbol{\Omega}$ can be huge, it is sparse and the sparse Cholesky decomposition has been studied extensively.
- We can do even better than the general approach to the Cholesky decomposition of sparse semidefinite matrices by taking advantage of the special structure of $\boldsymbol{Z}^\mathsf{T}\boldsymbol{Z}+\boldsymbol{\Omega}$.
- Although not shown here this decomposition allows us to formulate a general approach to an EM algorithm (actually ECME) for the optimization and, furthermore, we can evaluate the gradient and Hessian of the profiled objective

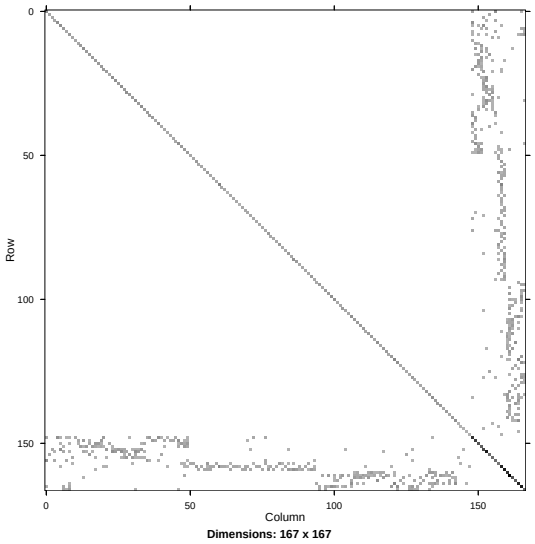
Symbolic analysis

- Sparse matrix methods often begin with a symbolic analysis to determine the locations of the non-zeros in the result.
- Although we will need to do the LDL decomposition for many different values of $\boldsymbol{\theta}$, we only need to do the symbolic analysis once.
- We can do the symbolic analysis on the $\boldsymbol{Z}^\mathsf{T}\boldsymbol{Z}$ matrix from the variance components model, even if we plan to fit a model with some $q_i > 1$. From the symbolic analysis of the variance components model and the values of the $q_i, i = 1,\dots,k$ we can determine the structure of $\boldsymbol{Z}^\mathsf{T}\boldsymbol{Z}$ and $\boldsymbol{L}$ for the more general model.

Fill-reducing permutation

- A crucial part of the symbolic analysis is determining a fill-reducing permutation of the rows and columns.
- In our case we consider only permutations of levels within groups.
- For the variance components model the blocks within groups are diagonal. Because the first block in $\boldsymbol{L}$ will also be diagonal but other blocks can be "filled-in", we order the factors by decreasing m_i.
- First factor is projected onto the other factors and a fill-reducing permutation of the second diagonal block is determined.
- This process is repeated for the remaining factors.
- Nested grouping factors, which do not need to have their levels permuted, are detected as part of this process.

ScotsSec variance components Z'Z



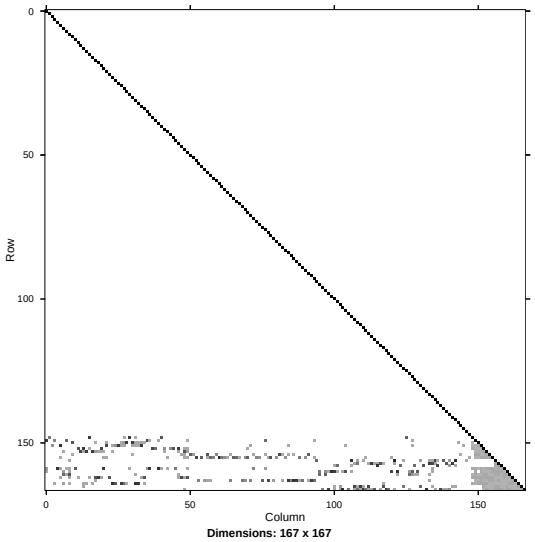
Fitting a model

```
> summary(fm1 <- lme(attain ~ verbal * sex, ScotsSec, ~1|primary + second))
Linear mixed-effects model fit by REML
Fixed: attain ~ verbal * sex
 Data: ScotsSec
      AIC      BIC    logLik
 14882.32 14925.32 -7434.162

Random effects:
 Groups   Name        Variance Std.Dev.
 primary  (Intercept) 0.275458 0.52484
 second   (Intercept) 0.014748 0.12144
 Residual             4.2531   2.0623
Fixed effects: Estimate Std. Error   DF t value Pr(>|t|)
(Intercept)  5.9147e+00 7.6795e-02 3431 77.0197  < 2e-16
verbal       1.5836e-01 3.7872e-03 3431 41.8136  < 2e-16
sexF         1.2155e-01 7.2413e-02 3431  1.6786  0.09332
verbal:sexF  2.5929e-03 5.3885e-03 3431  0.4812  0.63041

Number of Observations: 3435
Number of Groups: primary  second
                      148      19
```

ScotsSec variance components L



Model representation

```
> str(fm1@rep)
 list()
 - attr(*, "D")= num [1:167] 69.4 22.4 18.4 22.4 68.4 ...
 - attr(*, "Gp")= int [1:3] 0 148 167
 - attr(*, "Li")= int [1:434] 149 159 160 148 151 149 164 159 151 155 ...
 - attr(*, "Lp")= int [1:168] 0 3 4 5 7 8 12 16 19 25 ...
 - attr(*, "Lx")= num [1:434] 0.6480 0.1152 0.0144 0.3119 0.1627 ...
 - attr(*, "Omega")=List of 2
  ..$ primary: num [1, 1] 15.4
  ..$ second : num [1, 1] 288
 - attr(*, "Parent")= int [1:168] 149 148 151 149 159 151 153 149 156 150 ...
 - attr(*, "RXX")= num [1:5, 1:5] 0.0324 0.0000 0.0000 0.0000 0.0000 ...
 - attr(*, "RZX")= num [1:167, 1:5] 0.02192 0.00884 0.00446 0.00883 0.02038 ...
 - attr(*, "XtX")= num [1:5, 1:5] 3435    0    0    0    0 ...
 - attr(*, "ZtX")= num [1:167, 1:5] 54 7 3 7 53 55 22 15 33 18 ...
 - attr(*, "bVar")=List of 2
  ..$ primary: num [1, 1, 1:148] 0.125 0.212 0.233 0.212 0.127 ...
  ..$ second : num [1, 1, 1:19] 0.0547 0.0542 0.0548 0.0528 0.0519 ...
 - attr(*, "deviance")= num [1:2] 14843 14868
 - attr(*, "i")= int [1:470] 0 1 2 3 4 5 6 7 8 9 ...
 - attr(*, "nc")= int [1:4] 1 1 5 3435
 - attr(*, "p")= int [1:168] 0 1 2 3 4 5 6 7 8 9 ...
 - attr(*, "x")= num [1:470] 54 7 3 7 53 55 22 15 33 18 ...
```

ScotsSec example

- The fill-reducing permutation involves only the levels of *second* and is determined by a 19×19 matrix.
- In this case the fill-reducing is only moderately effective but it is not that important. For the Dallas data it is important.
- There are 470 non-redundant non-zeroes in $\boldsymbol{Z}^\top\boldsymbol{Z} + \boldsymbol{\Omega}$, 167 in $\boldsymbol{D}$, and 434 non-zero off-diagonals in $\boldsymbol{L}$ (the unit diagonal elements of $\boldsymbol{L}$ are not stored) for a total of 601. All other methods that use dense storage of the off-diagonal block would require at least 2979 elements to store $\boldsymbol{Z}^\top\boldsymbol{Z}$.
- The iteration process using a moderate number of ECME iterations followed by quasi-Newton optimization with analytic gradient is fast and stable.
- One can even fit large models to the Dallas data in a reasonable amount of time.

ScotsSec example (cont'd)

```
> system.time(lme(attain ~ verbal * sex, ScotsSec, ~1 | primary +
+     second, control = list(EMv = TRUE, msV = TRUE, opt = "optim",
+     niterEM = 14)))
  EM iterations
  0 14876.878  8.70355  67.7961
  1 14872.858  10.3663  81.3411
  2 14870.897  11.6441  94.3742
  3 14869.872  12.6156  106.565
  4 14869.309  13.3494  117.823
  5 14868.984  13.9012  128.164
  6 14868.788  14.3146  137.651
  7 14868.665  14.6234  146.360
  8 14868.585  14.8535  154.371
  9 14868.530  15.0243  161.758
 10 14868.491  15.1508  168.588
 11 14868.462  15.2442  174.920
 12 14868.440  15.3127  180.806
 13 14868.423  15.3629  186.290
 14 14868.409  15.3993  191.412
initial  value 14868.408855
final  value 14868.324922
converged
[1] 0.1 0.0 0.1 0.0 0.0
```

Fitting linear mixed models

- Specification of the model is straightforward.
  - Usual arguments for a model fitting function using formula, data, na.action, subset, etc. The formula specifies the response and the terms in the fixed-effects model.
  - The general form of the *random* argument is a named list of formulae (the names are those of the grouping factors). Short-cuts are provided for common cases.
  - **No** specification of nesting or crossing is needed.
  - **No** special forms of variance-covariance matrices are used (or needed - the one case that may be of practical use is handled by allowing grouping factors to be repeated).
- Method is fast and stable. With an analytic gradient (and Hessian) of the profiled criterion available, convergence can be reliably assessed. (In some cases finite estimates do not exist and the method should indicate this.)

Extensions

GLMM In a generalized linear mixed model the linear predictor is expressed as $\boldsymbol{X\beta} + \boldsymbol{Zb}$.

NLM In a nonlinear mixed model there is an underlying nonlinear model whose parameters each are expressed as $\boldsymbol{X\beta} + \boldsymbol{Zb}$ for (possibly different) model matrices $\boldsymbol{X}$ and $\boldsymbol{Z}$.

- In each case, for the model without random effects, there is an algorithm (IRLS or Gauss-Newton) that replaces that model by a linear least squares model and iterates.
- To get a first approximate solution to the mixed model estimates replace the least squares model by iterations of the penalized least squares problem for linear mixed models.
- These algorithms do not give MLEs for the GLMM nor for the NMM. However, they do get into a neighbourhood of the MLEs.

Extensions

- After convergence switch to optimization of a Laplacian or an adaptive Gauss-Hermite quadrature (AGQ) evaluation of the marginal likelihood.
- Both Laplace and AGQ require evaluation of the conditional modes, $\widehat{\boldsymbol{b}}(\boldsymbol{\theta}, \boldsymbol{\beta})$, for which the techniques we have described can be used.

General aspects of R

- R, and the community that supports it, provide **access** to state-of-the-art statistical computing and graphics
  - Available for all major operating systems
  - Available without regard to institutional wealth
  - Accessible to users of many levels of expertise (I use it in all my courses).
  - A platform for "reference implementations" of methods (see Brian Ripley's address to the RSS).
- R (and the S language in general) encourage an interactive approach to data analysis with heavy use of exploratory and presentation graphics.
- The S language provides a high-level modeling language in which you can concentrate on the model and not get bogged down in details (indicator variables, etc.)

R and multilevel modeling

- Here we provide access to a simple, effective, and general specification of linear mixed models and fast, space-efficient algorithms for determining parameter estimates. Combined with lattice graphics and other supporting technologies in R, I hope this will have a substantial impact on the practice of multilevel modeling.
- This research was supported by the U.S. Army Medical Research and Materiel Command under Contract No. DAMD17-02-C-0019. I would like to thank Deepayan Sarkar for his contributions to the code and his hours of patient listening, Harold Doran for his initial suggestion of using sparse matrix methods and Tim Davis for extensive email tutoring on sparse matrix methods.