

# Package ‘lmm’

April 17, 2009

**Version** 0.3-5

**Date** 2008-8-26

**Title** Linear mixed models

**Author** Original by Joseph L. Schafer <jls@stat.psu.edu>.

**Maintainer** Jing hua Zhao <jinghua.zhao@mrc-epid.cam.ac.uk>

**Depends** R (>= 2.0.0)

**Description** Some improved procedures for linear mixed models

**License** file LICENSE

**URL** <http://www.stat.psu.edu/~jls/misoftwa.html>

**Repository** CRAN

**Date/Publication** 2008-08-26 14:39:05

## R topics documented:

ecmehl.lmm . . . . .	2
ecmerml.lmm . . . . .	4
example.lmm . . . . .	7
fastmcmc.lmm . . . . .	10
fastml.lmm . . . . .	13
fastmode.lmm . . . . .	16
fastrml.lmm . . . . .	19
mgibbs.lmm . . . . .	22

<b>Index</b>	<b>25</b>
--------------	-----------

ecmeml.lmm

*ECME algorithm for maximum-likelihood (ML) estimation in linear mixed models***Description**

Computes ML estimates of parameters in linear mixed models using the ECME procedure described by Schafer (1998). This algorithm may be slow, requiring a large number of cycles to converge. In most cases, "fastml.lmm" will perform better. This function is provided mainly for comparison against "fastml.lmm".

For a description of the model, see the "Details" section below.

**Usage**

```
ecmeml.lmm(y, subj, pred, xcol, zcol, vmax, occ, start,
           maxits=1000, eps=0.0001)
```

**Arguments**

y	vector of responses. This is simply the individual $y_i$ vectors stacked upon one another. Each element of y represents the observed response for a particular subject-occasion, or for a particular unit within a cluster.
subj	vector of same length as y, giving the subject (or cluster) indicators $i$ for the elements of y. For example, suppose that y is $c(y_1, y_2, y_3, y_4)$ where $\text{length}(y_1)=2$ , $\text{length}(y_2)=3$ , $\text{length}(y_3)=2$ , and $\text{length}(y_4)=7$ . Then subj should be $c(1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 4, 4, 4, 4)$ .
pred	matrix of covariates used to predict y. The number of rows should be $\text{length}(y)$ . The first column will typically be constant (one), and the remaining columns correspond to other variables appearing in $X_i$ and $Z_i$ .
xcol	vector of integers indicating which columns of pred will be used in $X_i$ . That is, $\text{pred}[,xcol]$ is the $X_i$ matrices (stacked upon one another).
zcol	vector of integers indicating which columns of pred will be used in $Z_i$ . That is, $\text{pred}[,zcol]$ is the $Z_i$ matrices (stacked upon one another).
vmax	optional matrix of dimension $c(\max(\text{occ}), \max(\text{occ}))$ from which the $V_i$ matrices will be extracted. In a longitudinal dataset, vmax would represent the $V_i$ matrix for an individual with responses at all possible occasions $1, 2, \dots, n_{\max} = \max(\text{occ})$ ; for individuals with responses at only a subset of these occasions, the $V_i$ will be obtained by extracting the rows and columns of vmax for those occasions. If no vmax is specified by the user, an identity matrix is used. In most applications of this model one will want to have $V_i = \text{identity}$ , so most of the time this argument can be omitted.
occ	vector of same length as y indicating the "occasions" for the elements of y. This argument is relevant only if a non-identity vmax is specified. In a longitudinal dataset where each individual is measured on at most $n_{\max}$ distinct occasions, each element of y corresponds to one subject-occasion, and the elements of occ should be coded as $1, 2, \dots, n_{\max}$ to indicate these occasion labels. (You should

	label the occasions as 1,2,...,nmax even if they are not equally spaced in time; the actual times of measurement will be incorporated into the matrix "pred".)
start	optional starting values of the parameters. If this argument is not given then the function chooses its own starting values. This argument should be a list of three elements named "beta", "psi", and "sigma2". Note that "beta" should be a vector of the same length as "xcol", "psi" should be a matrix of dimension $c(\text{length}(zcol), \text{length}(zcol))$ , and "sigma2" should be a scalar.
maxits	maximum number of cycles to be performed. The algorithm runs to convergence or until "maxits" iterations, whichever comes first.
eps	convergence criterion. The algorithm is considered to have converged if the relative differences in all parameters from one iteration to the next are less than eps—that is, if $\text{all}(\text{abs}(\text{new-old}) < \text{eps} * \text{abs}(\text{old}))$ .

### Details

For details of the algorithm, see Section 3 of Schafer (1998).

The model, which is typically applied to longitudinal or clustered responses, is

$$y_i = X_i \beta + Z_i b_i + e_i, \quad i=1, \dots, m,$$

where

$y_i = (n_i \times 1)$  response vector for subject or cluster  $i$ ;  $X_i = (n_i \times p)$  matrix of covariates;  $Z_i = (n_i \times q)$  matrix of covariates;  $\beta = (p \times 1)$  vector of coefficients common to the population (fixed effects);  $b_i = (q \times 1)$  vector of coefficients specific to subject or cluster  $i$  (random effects); and  $e_i = (n_i \times 1)$  vector of residual errors.

The vector  $b_i$  is assumed to be normally distributed with mean zero and unstructured covariance matrix  $\psi$ ,

$$b_i \sim N(0, \psi) \text{ independently for } i=1, \dots, m.$$

The residual vector  $e_i$  is assumed to be

$$e_i \sim N(0, \sigma^2 V_i)$$

where  $V_i$  is a known  $(n_i \times n_i)$  matrix. In most applications,  $V_i$  is the identity matrix.

### Value

a list containing the following components.

beta	vector of same length as "xcol" containing estimated fixed effects.
sigma2	estimate of residual error variance.
psi	matrix of dimension $c(\text{length}(zcol), \text{length}(zcol))$ containing estimated variances and covariances of the random effects.
converged	T if the algorithm converged, F if it did not.
iter	number of iterations actually performed. Will be equal to "maxits" if converged=F.
loglik	vector of length "iter" reporting the value of the loglikelihood at each iteration.

<code>cov.beta</code>	matrix of dimension $c(\text{length}(xcol), \text{length}(xcol))$ containing estimated variances and covariances for elements of "beta". These are conventional estimates which regard the variance parameters ( $\sigma^2$ and $\psi$ ) as fixed at their ML estimates.
<code>b.hat</code>	a matrix with $\text{length}(zcol)$ rows and $m$ columns, where $b.hat[,i]$ is an empirical Bayes estimate of $b_i$ .
<code>cov.b</code>	an array of dimension $\text{length}(zcol)$ by $\text{length}(zcol)$ by $m$ , where $cov.b[,i]$ is an empirical Bayes estimate of the covariance matrix associated with $b_i$ . These are conventional estimates which regard the variance parameters ( $\sigma^2$ and $\psi$ ) as fixed at their ML estimates. (An improved version which incorporates variance-parameter uncertainty is available from the function "fastrml.lmm".)

## References

Schafer, J.L. (1998) Some improved procedures for linear mixed models. Submitted to Journal of the American Statistical Association.

## See Also

[ecmerml.lmm](#), [fastml.lmm](#), [fastrml.lmm](#), [fastmode.lmm](#), [mgibbs.lmm](#), [fastmcmc.lmm](#), [example.lmm](#)

## Examples

```
## Not run:
For a detailed example, see the file "example.lmm.R" distributed
with this library.
## End(Not run)
```

---

<code>ecmerml.lmm</code>	<i>ECME algorithm for restricted maximum-likelihood (RML) estimation in linear mixed models</i>
--------------------------	---

---

## Description

Computes RML estimates of parameters in linear mixed models using the ECME procedure described by Schafer (1998). This algorithm may be slow, requiring a large number of cycles to converge. In most cases, "fastrml.lmm" will perform better. This function is provided mainly for comparison against "fastrml.lmm".

For a description of the model, see the "Details" section below.

## Usage

```
ecmerml.lmm(y, subj, pred, xcol, zcol, vmax, occ, start,
            maxits=1000, eps=0.0001)
```

**Arguments**

<code>y</code>	vector of responses. This is simply the individual $y_i$ vectors stacked upon one another. Each element of $y$ represents the observed response for a particular subject-occasion, or for a particular unit within a cluster.
<code>subj</code>	vector of same length as $y$ , giving the subject (or cluster) indicators $i$ for the elements of $y$ . For example, suppose that $y$ is $c(y_1, y_2, y_3, y_4)$ where $\text{length}(y_1)=2$ , $\text{length}(y_2)=3$ , $\text{length}(y_3)=2$ , and $\text{length}(y_4)=7$ . Then <code>subj</code> should be $c(1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 4, 4, 4, 4)$ .
<code>pred</code>	matrix of covariates used to predict $y$ . The number of rows should be $\text{length}(y)$ . The first column will typically be constant (one), and the remaining columns correspond to other variables appearing in $X_i$ and $Z_i$ .
<code>xcol</code>	vector of integers indicating which columns of <code>pred</code> will be used in $X_i$ . That is, <code>pred[,xcol]</code> is the $X_i$ matrices (stacked upon one another).
<code>zcol</code>	vector of integers indicating which columns of <code>pred</code> will be used in $Z_i$ . That is, <code>pred[,zcol]</code> is the $Z_i$ matrices (stacked upon one another).
<code>vmax</code>	optional matrix of dimension $c(\max(\text{occ}), \max(\text{occ}))$ from which the $V_i$ matrices will be extracted. In a longitudinal dataset, <code>vmax</code> would represent the $V_i$ matrix for an individual with responses at all possible occasions $1, 2, \dots, n_{\max} = \max(\text{occ})$ ; for individuals with responses at only a subset of these occasions, the $V_i$ will be obtained by extracting the rows and columns of <code>vmax</code> for those occasions. If no <code>vmax</code> is specified by the user, an identity matrix is used. In most applications of this model one will want to have $V_i = \text{identity}$ , so most of the time this argument can be omitted.
<code>occ</code>	vector of same length as $y$ indicating the "occasions" for the elements of $y$ . This argument is relevant only if a non-identity <code>vmax</code> is specified. In a longitudinal dataset where each individual is measured on at most $n_{\max}$ distinct occasions, each element of $y$ corresponds to one subject-occasion, and the elements of <code>occ</code> should be coded as $1, 2, \dots, n_{\max}$ to indicate these occasion labels. (You should label the occasions as $1, 2, \dots, n_{\max}$ even if they are not equally spaced in time; the actual times of measurement will be incorporated into the matrix "pred".)
<code>start</code>	optional starting values of the parameters. If this argument is not given then the function chooses its own starting values. This argument should be a list of three elements named "beta", "psi", and "sigma2". Note that "beta" should be a vector of the same length as "xcol", "psi" should be a matrix of dimension $c(\text{length}(zcol), \text{length}(zcol))$ , and "sigma2" should be a scalar.
<code>maxits</code>	maximum number of cycles to be performed. The algorithm runs to convergence or until "maxits" iterations, whichever comes first.
<code>eps</code>	convergence criterion. The algorithm is considered to have converged if the relative differences in all parameters from one iteration to the next are less than <code>eps</code> —that is, if $\text{all}(\text{abs}(\text{new}-\text{old}) < \text{eps} * \text{abs}(\text{old}))$ .

**Details**

For details of the algorithm, see Section 3 of Schafer (1998).

The model, which is typically applied to longitudinal or clustered responses, is

$$y_i = X_i \beta + Z_i b_i + e_i, \quad i=1, \dots, m,$$

where

$y_i = (n_i \times 1)$  response vector for subject or cluster  $i$ ;  $X_i = (n_i \times p)$  matrix of covariates;  $Z_i = (n_i \times q)$  matrix of covariates;  $\beta = (p \times 1)$  vector of coefficients common to the population (fixed effects);  $b_i = (q \times 1)$  vector of coefficients specific to subject or cluster  $i$  (random effects); and  $e_i = (n_i \times 1)$  vector of residual errors.

The vector  $b_i$  is assumed to be normally distributed with mean zero and unstructured covariance matrix  $\psi_i$ ,

$b_i \sim N(0, \psi_i)$  independently for  $i=1, \dots, m$ .

The residual vector  $e_i$  is assumed to be

$e_i \sim N(0, \sigma^2 V_i)$

where  $V_i$  is a known  $(n_i \times n_i)$  matrix. In most applications,  $V_i$  is the identity matrix.

### Value

a list containing the following components.

<code>beta</code>	vector of same length as "xcol" containing estimated fixed effects.
<code>sigma2</code>	estimate of residual error variance.
<code>psi</code>	matrix of dimension $c(\text{length}(zcol), \text{length}(zcol))$ containing estimated variances and covariances of the random effects.
<code>converged</code>	T if the algorithm converged, F if it did not.
<code>iter</code>	number of iterations actually performed. Will be equal to "maxits" if converged=F.
<code>loglik</code>	vector of length "iter" reporting the value of the "restricted" loglikelihood at each iteration.
<code>cov.beta</code>	matrix of dimension $c(\text{length}(xcol), \text{length}(xcol))$ containing estimated variances and covariances for elements of "beta". These are conventional estimates which regard the variance parameters ( <code>sigma2</code> and <code>psi</code> ) as fixed at their RML estimates.
<code>b.hat</code>	a matrix with $\text{length}(zcol)$ rows and $m$ columns, where <code>b.hat[,i]</code> is an empirical Bayes estimate of $b_i$ .
<code>cov.b</code>	an array of dimension $\text{length}(zcol)$ by $\text{length}(zcol)$ by $m$ , where <code>cov.b[,i]</code> is an empirical Bayes estimate of the covariance matrix associated with $b_i$ . These are conventional estimates which regard the variance parameters ( <code>sigma2</code> and <code>psi</code> ) as fixed at their RML estimates. (An improved version which incorporates variance-parameter uncertainty is available from the function "fastrml.lmm".)

### References

Schafer, J.L. (1998) Some improved procedures for linear mixed models. Submitted to Journal of the American Statistical Association.

### See Also

[ecmeml.lmm](#), [fastml.lmm](#), [fastrml.lmm](#), [fastmode.lmm](#), [mgibbs.lmm](#), [fastmcmc.lmm](#), [example.lmm](#)

**Examples**

```
## Not run:
For a detailed example, see the file "example.lmm.R" distributed
with this library.

## End(Not run)
```

---

example.lmm

*lmm library example command file*


---

**Description**

These data, taken from an article by Weil, A.T., Zinberg, N.E. and Nelson, J.M. (1968; Clinical and psychological effects of marihuana in man; Science, 162, 1234-1242), come from a pilot study to investigate the clinical and psychological effects of marijuana use in human subjects. Nine subjects each received three treatments—low-dose, high-dose, and placebo. Under each treatment, changes in heart rate (beats per minute) were measured 15 and 90 minutes after administration. NA denotes a missing value.

**See Also**

[ecmml.lmm](#), [ecmerml.lmm](#), [fastml.lmm](#), [fastrml.lmm](#), [fastmcmc.lmm](#), [fastmode.lmm](#), [mgibbs.lmm](#)

**Examples**

```
#
# -----
#               15 minutes                90 minutes
#               -----                -----
#               Placebo  Low  High      Placebo  Low  High
# -----
# Subject 1      16    20   16          20    -6   -4
#              2     12   24   12          -6     4   -8
#              3      8     8   26          -4     4    8
#              4     20     8   NA          NA    20   -4
#              5      8     4   -8          NA    22   -8
#              6     10    20   28          -20   -4   -4
#              7      4    28   24           12    8   18
#              8     -8    20   24           -3    8  -24
#              9      NA    20   24            8   12   NA
# -----
#
#####
# Below we show how to fit a traditional compound symmetry model
# with a fixed effect for each column (occasion) and a random
# intercept for each subject. First we enter the data.
#
y <- c(16,20,16,20,-6,-4,
```

```

12,24,12,-6,4,-8,
8,8,26,-4,4,8,
20,8,20,-4,
8,4,-8,22,-8,
10,20,28,-20,-4,-4,
4,28,24,12,8,18,
-8,20,24,-3,8,-24,
20,24,8,12)
occ <- c(1,2,3,4,5,6,
1,2,3,4,5,6,
1,2,3,4,5,6,
1,2,5,6,
1,2,3,5,6,
1,2,3,4,5,6,
1,2,3,4,5,6,
1,2,3,4,5,6,
2,3,4,5)
subj <- c(1,1,1,1,1,1,
2,2,2,2,2,2,
3,3,3,3,3,3,
4,4,4,4,
5,5,5,5,5,
6,6,6,6,6,6,
7,7,7,7,7,7,
8,8,8,8,8,8,
9,9,9,9)
#####
# Now we must specify the model.
# If the six measurements per subject were ordered in time, we might
# consider using a model with time of measurement entered with linear
# (or perhaps higher-order polynomial) effects. But because the
# six measurements are not clearly ordered, let's use a model that has
# an intercept and five dummy codes to allow the population means for
# the six occasions to be estimated freely. We will also allow the
# intercept to randomly vary by subject. For a subject i with no
# missing values, the covariate matrices will be
#
#
#           1 1 0 0 0 0           1
#           1 0 1 0 0 0           1
#           Xi = 1 0 0 1 0 0           Zi = 1
#           1 0 0 0 1 0           1
#           1 0 0 0 0 1           1
#           1 0 0 0 0 0           1
#
# The Xi's and Zi's are combined into a single matrix called
# pred. The pred matrix has length(y) rows. Each column of Xi and Zi
# must be represented in pred. Because Zi is merely the first column
# of Xi, we do not need to enter that column twice. So pred is simply
# the matrices Xi (i=1,...,9), stacked upon each other.
#
pred <- cbind(int=rep(1,49),dummy1=1*(occ==1),dummy2=1*(occ==2),
dummy3=1*(occ==3),dummy4=1*(occ==4),dummy5=1*(occ==5))
xcol <- 1:6

```

```

zcol <- 1
#####
# Now find ML estimates using the ECME procedure and the faster
# scoring algorithm
#
ecmestml.result <- ecmestml.lmm(y,subj,pred,xcol,zcol)
fastml.result <- fastml.lmm(y,subj,pred,xcol,zcol)
#
# In this example, ECME converged in 212 cycles, but the fast
# algorithm took only 8. The results can be viewed by printing the
# various components of "ecmestml.result" and "fastml.result".
# For example, extract the ML estimate of the fixed effects beta.
#
beta.hat <- fastml.result$beta
#
# Because of the dummy codes used in the Xi's, the first element of
# beta (the intercept) estimates the mean for the last occasion,
# and the other elements of beta estimate the differences in means
# between the first five occasions and the last one. So we can find
# the estimated means for the six occasions like this:
#
muhat <- c(beta.hat[2]+beta.hat[1], beta.hat[3]+beta.hat[1],
           beta.hat[4]+beta.hat[1], beta.hat[5]+beta.hat[1],
           beta.hat[6]+beta.hat[1], beta.hat[1])
#
# The functions for RML estimation work exactly the same way:
#
ecmerml.result <- ecmerml.lmm(y,subj,pred,xcol,zcol)
fastrml.result <- fastrml.lmm(y,subj,pred,xcol,zcol)
#
#####
# The function "fastrml.lmm" calculates the improved variance
# estimates for random effects described in Section 4 of Schafer
# (1998). The code below reproduces Table 2, which compares
# 95% interval estimates under the new method to conventional
# empirical Bayes intervals.
#
b.hat <- as.vector(fastrml.result$b.hat)
se.new <- sqrt(as.vector(fastrml.result$cov.b.new))
se.old <- sqrt(as.vector(fastrml.result$cov.b))
table2 <- cbind(round(b.hat,3),
               round(cbind(b.hat-2*se.old,b.hat+2*se.old,
                           b.hat-2*se.new,b.hat+2*se.new),2),
               round(100*(se.new-se.old)/se.old))
dimnames(table2) <- list(paste("Subject",format(1:9)),
                       c("Est.", "Lower.old", "Upper.old", "Lower.new", "Upper.new",
                         "Increase (%)"))
print(table2)
#
#####
# The functions "mgibbs.lmm" and "fastmcmc.lmm" perform the MCMC
# procedures described in Section 5. The code below runs each
# algorithm for 5,000 cycles, and then displays autocorrelation

```

```

# plots like those of Figure 1.
#
prior <- list(a=3*100,b=3,c=3,Dinv=3*5)
gibbs.result <- mgibbs.lmm(y,subj,pred,xcol,zcol,prior=prior,
  seed=1234,iter=5000)
fcmc.result <- fastmcmc.lmm(y,subj,pred,xcol,zcol,prior=prior,
  seed=2345,iter=5000)
#
# Before doing this, make sure that a graphics device is active:
# library(ts)
par(mfrow=c(2,1))
#
# there were problems with debian Linux; so add ylim
acf(log(gibbs.result$psi.series[1,1,]),lag.max=10, ylim=0:1)
acf(log(fcmc.result$psi.series[1,1,]),lag.max=10, ylim=0:1)

#####

```

---

fastmcmc.lmm	<i>Rapidly converging Markov chain Monte Carlo algorithm for Bayesian inference in linear mixed models</i>
--------------	--

---

## Description

Simulates posterior draws of parameters in linear mixed models using the rapidly converging Markov chain Monte Carlo (MCMC) procedure described by Schafer (1998), which combines a Metropolis-Hastings algorithm with a modified Gibbs sampler.

Prior to the MCMC simulation, the posterior mode of the variance parameters is found using the algorithm of "fastmode.lmm". The results from a call to "fastmode.lmm" are returned along with the MCMC results.

For a description of the model and the prior distribution, see the "Details" section below.

## Usage

```

fastmcmc.lmm(y, subj, pred, xcol, zcol, prior, seed, vmax,
  occ, start.mode, maxits=100, eps=0.0001, iter=1000,
  start.mcmc, df=4)

```

## Arguments

y	vector of responses. This is simply the individual $y_i$ vectors stacked upon one another. Each element of y represents the observed response for a particular subject-occasion, or for a particular unit within a cluster.
subj	vector of same length as y, giving the subject (or cluster) indicators $i$ for the elements of y. For example, suppose that y is $c(y_1, y_2, y_3, y_4)$ where $\text{length}(y_1)=2$ , $\text{length}(y_2)=3$ , $\text{length}(y_3)=2$ , and $\text{length}(y_4)=7$ . Then subj should be $c(1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 4, 4, 4)$ .

<code>pred</code>	matrix of covariates used to predict $y$ . The number of rows should be <code>length(y)</code> . The first column will typically be constant (one), and the remaining columns correspond to other variables appearing in $X_i$ and $Z_i$ .
<code>xcol</code>	vector of integers indicating which columns of <code>pred</code> will be used in $X_i$ . That is, <code>pred[,xcol]</code> is the $X_i$ matrices (stacked upon one another).
<code>zcol</code>	vector of integers indicating which columns of <code>pred</code> will be used in $Z_i$ . That is, <code>pred[,zcol]</code> is the $Z_i$ matrices (stacked upon one another).
<code>prior</code>	A list with four components specifying the hyperparameters of the prior distribution applied to $\sigma^2$ and $\psi$ . The components must be named "a", "b", "c", and "Dinv". All are scalars except for "Dinv", which is a matrix of dimension <code>c(length(zcol),length(zcol))</code> .
<code>seed</code>	Seed for random number generator. This should be a positive integer.
<code>vmax</code>	optional matrix of dimension <code>c(max(occ),max(occ))</code> from which the $V_i$ matrices will be extracted. In a longitudinal dataset, <code>vmax</code> would represent the $V_i$ matrix for an individual with responses at all possible occasions $1, 2, \dots, n_{\max} = \max(\text{occ})$ ; for individuals with responses at only a subset of these occasions, the $V_i$ will be obtained by extracting the rows and columns of <code>vmax</code> for those occasions. If no <code>vmax</code> is specified by the user, an identity matrix is used. In most applications of this model one will want to have $V_i = \text{identity}$ , so most of the time this argument can be omitted.
<code>occ</code>	vector of same length as $y$ indicating the "occasions" for the elements of $y$ . This argument is relevant only if a non-identity <code>vmax</code> is specified. In a longitudinal dataset where each individual is measured on at most $n_{\max}$ distinct occasions, each element of $y$ corresponds to one subject-occasion, and the elements of <code>occ</code> should be coded as $1, 2, \dots, n_{\max}$ to indicate these occasion labels. (You should label the occasions as $1, 2, \dots, n_{\max}$ even if they are not equally spaced in time; the actual times of measurement will be incorporated into the matrix "pred".)
<code>start.mode</code>	optional starting values of the parameters for the mode-finding procedure. If this argument is not given then the function chooses its own starting values. This argument should be a list of three elements named "beta", "psi", and "sigma2". Note that "beta" should be a vector of the same length as "xcol", "psi" should be a matrix of dimension <code>c(length(zcol),length(zcol))</code> , and "sigma2" should be a scalar.
<code>maxits</code>	maximum number of cycles of the mode-finding procedure. The algorithm runs to convergence or until "maxits" iterations, whichever comes first.
<code>eps</code>	convergence criterion for the mode-finding procedure. The algorithm is considered to have converged if the relative differences in all parameters from one iteration to the next are less than <code>eps</code> —that is, if <code>all(abs(new-old)&lt;eps*abs(old))</code> .
<code>iter</code>	number of cycles of the MCMC procedure to be performed.
<code>start.mcmc</code>	optional starting values of the parameters for the MCMC procedure. If this argument is not given, then the procedure is started at the posterior mode.
<code>df</code>	degrees of freedom for the multivariate $t$ approximation in the Metropolis-Hastings algorithm.

## Details

The algorithm is described in Section 5 of Schafer (1998).

The model, which is typically applied to longitudinal or clustered responses, is

$$y_i = X_i \beta + Z_i b_i + e_i, \quad i=1, \dots, m,$$

where

$y_i = (n_i \times 1)$  response vector for subject or cluster  $i$ ;  $X_i = (n_i \times p)$  matrix of covariates;  $Z_i = (n_i \times q)$  matrix of covariates;  $\beta = (p \times 1)$  vector of coefficients common to the population (fixed effects);  $b_i = (q \times 1)$  vector of coefficients specific to subject or cluster  $i$  (random effects); and  $e_i = (n_i \times 1)$  vector of residual errors.

The vector  $b_i$  is assumed to be normally distributed with mean zero and unstructured covariance matrix  $\psi_i$ ,

$$b_i \sim N(0, \psi_i) \text{ independently for } i=1, \dots, m.$$

The residual vector  $e_i$  is assumed to be

$$e_i \sim N(0, \sigma^2 V_i)$$

where  $V_i$  is a known  $(n_i \times n_i)$  matrix. In most applications,  $V_i$  is the identity matrix.

The prior distribution applied to the within-unit residual variance is scaled inverted-chisquare,

$$\sigma^2 \sim a / \text{chisq}(b),$$

where  $\text{chisq}(b)$  denotes a chisquare random variable with  $b$  degrees of freedom, and  $a$  and  $b$  are user-defined hyperparameters. Values for the hyperparameters may be chosen by regarding  $a/b$  as a rough prior guess for  $\sigma^2$ , and as the imaginary degrees of freedom on which this guess is based.

The prior distribution applied to the between-unit covariance matrix is inverted Wishart,

$$\psi_i^{-1} \sim W(c, D),$$

where  $\psi_i^{-1}$  is the inverse of the between-unit covariance matrix  $\psi_i$ , and  $W(c, D)$  denotes a Wishart distribution with degrees of freedom  $c$  and scale matrix  $D$ . Values for the hyperparameters may be chosen by regarding  $D \text{inv}/c$  (the inverse of  $D$  divided by  $c$ ) as a rough prior guess for  $\psi_i$ , and  $c$  as the imaginary degrees of freedom on which this guess is based.

An improper uniform prior density function is applied to the fixed effects  $\beta$ .

## Value

a list containing the following components.

<code>beta</code>	simulated value of coefficients $\beta$ after "iter" cycles of the MCMC algorithm. This is a vector of the same length as <code>xcol</code> .
<code>sigma2</code>	simulated value of the residual variance $\sigma^2$ after "iter" cycles of the MCMC algorithm.
<code>psi</code>	simulated value of the between-unit covariance matrix $\psi_i$ after "iter" cycles of the MCMC algorithm.
<code>sigma2.series</code>	vector of length "iter" containing the entire history of simulated values of $\sigma^2$ . That is, <code>sigma2.series[t]</code> contains the value of $\sigma^2$ at cycle $t$ .

<code>psi.series</code>	array of dimension $c(\text{length}(\text{zcol}), \text{length}(\text{zcol}), \text{iter})$ containing the entire history of simulated values of $\psi$ . That is, <code>psi.series[,t]</code> contains the value of $\psi$ at cycle $t$ .
<code>ratios</code>	vector of length "iter" containing the entire history of acceptance ratios from the Metropolis-Hastings algorithm. These ratios diagnose the quality of the multivariate $t$ approximation. If the approximation were perfect, all of these ratios would be equal to one.
<code>reject</code>	logical vector of length "iter" indicating, for each cycle of the algorithm, whether the Metropolis-Hastings candidate was accepted (T) or rejected (F).
<code>mode.list</code>	a list containing the results of the mode-finding procedure. The contents of this list are identical to those produced by "fastmode.lmm". For more information, see the help file for "fastmode.lmm".

## References

Schafer, J.L. (1998) Some improved procedures for linear mixed models. Submitted to Journal of the American Statistical Association.

## See Also

[ecmest.lmm](#), [ecmerml.lmm](#), [fastml.lmm](#), [fastrml.lmm](#), [fastmode.lmm](#), [mgibbs.lmm](#), [example.lmm](#)

## Examples

```
## Not run:
For a detailed example, see the file "example.lmm.R" distributed
with this library.

## End(Not run)
```

---

<code>fastml.lmm</code>	<i>Rapidly converging algorithm for maximum-likelihood (ML) estimation in linear mixed models</i>
-------------------------	---

---

## Description

Computes ML estimates of parameters in linear mixed models using the rapidly converging procedure described by Schafer (1998), which combines Fisher scoring with an ECME algorithm.

For a description of the model, see the "Details" section below.

## Usage

```
fastml.lmm(y, subj, pred, xcol, zcol, vmax, occ, start,
           maxits=50, eps=0.0001)
```

**Arguments**

<code>y</code>	vector of responses. This is simply the individual $y_i$ vectors stacked upon one another. Each element of $y$ represents the observed response for a particular subject-occasion, or for a particular unit within a cluster.
<code>subj</code>	vector of same length as $y$ , giving the subject (or cluster) indicators $i$ for the elements of $y$ . For example, suppose that $y$ is $c(y_1, y_2, y_3, y_4)$ where $\text{length}(y_1)=2$ , $\text{length}(y_2)=3$ , $\text{length}(y_3)=2$ , and $\text{length}(y_4)=7$ . Then <code>subj</code> should be $c(1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 4, 4, 4, 4)$ .
<code>pred</code>	matrix of covariates used to predict $y$ . The number of rows should be $\text{length}(y)$ . The first column will typically be constant (one), and the remaining columns correspond to other variables appearing in $X_i$ and $Z_i$ .
<code>xcol</code>	vector of integers indicating which columns of <code>pred</code> will be used in $X_i$ . That is, <code>pred[,xcol]</code> is the $X_i$ matrices (stacked upon one another).
<code>zcol</code>	vector of integers indicating which columns of <code>pred</code> will be used in $Z_i$ . That is, <code>pred[,zcol]</code> is the $Z_i$ matrices (stacked upon one another).
<code>vmax</code>	optional matrix of dimension $c(\text{max}(\text{occ}), \text{max}(\text{occ}))$ from which the $V_i$ matrices will be extracted. In a longitudinal dataset, <code>vmax</code> would represent the $V_i$ matrix for an individual with responses at all possible occasions $1, 2, \dots, \text{nmax}=\text{max}(\text{occ})$ ; for individuals with responses at only a subset of these occasions, the $V_i$ will be obtained by extracting the rows and columns of <code>vmax</code> for those occasions. If no <code>vmax</code> is specified by the user, an identity matrix is used. In most applications of this model one will want to have $V_i = \text{identity}$ , so most of the time this argument can be omitted.
<code>occ</code>	vector of same length as $y$ indicating the "occasions" for the elements of $y$ . This argument is relevant only if a non-identity <code>vmax</code> is specified. In a longitudinal dataset where each individual is measured on at most <code>nmax</code> distinct occasions, each element of $y$ corresponds to one subject-occasion, and the elements of <code>occ</code> should be coded as $1, 2, \dots, \text{nmax}$ to indicate these occasion labels. (You should label the occasions as $1, 2, \dots, \text{nmax}$ even if they are not equally spaced in time; the actual times of measurement will be incorporated into the matrix "pred".)
<code>start</code>	optional starting values of the parameters. If this argument is not given then the function chooses its own starting values. This argument should be a list of three elements named "beta", "psi", and "sigma2". Note that "beta" should be a vector of the same length as "xcol", "psi" should be a matrix of dimension $c(\text{length}(zcol), \text{length}(zcol))$ , and "sigma2" should be a scalar.
<code>maxits</code>	maximum number of cycles to be performed. The algorithm runs to convergence or until "maxits" iterations, whichever comes first.
<code>eps</code>	convergence criterion. The algorithm is considered to have converged if the relative differences in all parameters from one iteration to the next are less than <code>eps</code> —that is, if $\text{all}(\text{abs}(\text{new}-\text{old}) < \text{eps} * \text{abs}(\text{old}))$ .

**Details**

A full description of the algorithm is given in Section 3 of Schafer (1998). Scoring is carried out on  $\log(\text{sigma}2)$  and the nonredundant elements of the inverse of  $\text{psi}/\text{sigma}2$ , taking logs of the diagonal elements.

The model, which is typically applied to longitudinal or clustered responses, is

$$y_i = X_i \beta + Z_i b_i + e_i, \quad i=1, \dots, m,$$

where

$y_i = (n_i \times 1)$  response vector for subject or cluster  $i$ ;  $X_i = (n_i \times p)$  matrix of covariates;  $Z_i = (n_i \times q)$  matrix of covariates;  $\beta = (p \times 1)$  vector of coefficients common to the population (fixed effects);  $b_i = (q \times 1)$  vector of coefficients specific to subject or cluster  $i$  (random effects); and  $e_i = (n_i \times 1)$  vector of residual errors.

The vector  $b_i$  is assumed to be normally distributed with mean zero and unstructured covariance matrix  $\psi_i$ ,

$$b_i \sim N(0, \psi_i) \text{ independently for } i=1, \dots, m.$$

The residual vector  $e_i$  is assumed to be

$$e_i \sim N(0, \sigma^2 V_i)$$

where  $V_i$  is a known  $(n_i \times n_i)$  matrix. In most applications,  $V_i$  is the identity matrix.

## Value

a list containing the following components.

<code>beta</code>	vector of same length as "xcol" containing estimated fixed effects.
<code>sigma2</code>	estimate of residual error variance.
<code>psi</code>	matrix of dimension $c(\text{length}(zcol), \text{length}(zcol))$ containing estimated variances and covariances of the random effects.
<code>converged</code>	T if the algorithm converged, F if it did not.
<code>iter</code>	number of iterations actually performed. Will be equal to "maxits" if converged=F.
<code>reject</code>	a logical vector of length <code>iter</code> indicating, for each iteration, whether the scoring estimates were rejected and replaced by ECME estimates (T), or whether the scoring estimates were accepted (F). Scoring estimates are rejected if they do not increase the loglikelihood.
<code>loglik</code>	vector of length "iter" reporting the value of the loglikelihood at each iteration.
<code>cov.beta</code>	matrix of dimension $c(\text{length}(xcol), \text{length}(xcol))$ containing estimated variances and covariances for elements of "beta". These are conventional estimates which regard the variance parameters ( <code>sigma2</code> and <code>psi</code> ) as fixed at their ML estimates.
<code>b.hat</code>	a matrix with <code>length(zcol)</code> rows and <code>m</code> columns, where <code>b.hat[,i]</code> is an empirical Bayes estimate of $b_i$ .
<code>cov.b</code>	an array of dimension <code>length(zcol)</code> by <code>length(zcol)</code> by <code>m</code> , where <code>cov.b[,i]</code> is an empirical Bayes estimate of the covariance matrix associated with $b_i$ . These are conventional estimates which regard the variance parameters ( <code>sigma2</code> and <code>psi</code> ) as fixed at their ML estimates. (An improved version which incorporates variance-parameter uncertainty is available from the function "fastml.lmm".)

## References

Schafer, J.L. (1998) Some improved procedures for linear mixed models. Submitted to Journal of the American Statistical Association.

**See Also**

[ecmml.lmm](#), [ecmerml.lmm](#), [fastrml.lmm](#), [fastmode.lmm](#), [mgibbs.lmm](#), [fastmcmc.lmm](#), [example.lmm](#)

**Examples**

```
## Not run:
For a detailed example, see the file "example.lmm.R" distributed
with this library.

## End(Not run)
```

---

fastmode.lmm	<i>Rapidly converging algorithm for calculating posterior modes in linear mixed models</i>
--------------	--

---

**Description**

Computes the marginal posterior mode of the variance parameters in linear mixed models using a rapidly converging procedure described by Schafer (1998), which combines Fisher scoring with an ECME algorithm. The method is a minor modification of the restricted maximum-likelihood (RML) procedure used in "fastrml.lmm". The model is identical to that of "fastrml.lmm" with the addition of prior distributions for the variance parameters.

For a description of the prior distribution, see the "Details" section below.

**Usage**

```
fastmode.lmm(y, subj, pred, xcol, zcol, prior, vmax, occ, start,
             maxits=100, eps=0.0001)
```

**Arguments**

Identical to those for the function "fastrml.lmm", with one additional required argument:

y	vector of responses. This is simply the individual $y_i$ vectors stacked upon one another. Each element of y represents the observed response for a particular subject-occasion, or for a particular unit within a cluster.
subj	vector of same length as y, giving the subject (or cluster) indicators $i$ for the elements of y. For example, suppose that y is $c(y_1, y_2, y_3, y_4)$ where $\text{length}(y_1)=2$ , $\text{length}(y_2)=3$ , $\text{length}(y_3)=2$ , and $\text{length}(y_4)=7$ . Then subj should be $c(1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 4, 4, 4, 4)$ .
pred	matrix of covariates used to predict y. The number of rows should be $\text{length}(y)$ . The first column will typically be constant (one), and the remaining columns correspond to other variables appearing in $X_i$ and $Z_i$ .
xcol	vector of integers indicating which columns of pred will be used in $X_i$ . That is, $\text{pred}[,xcol]$ is the $X_i$ matrices (stacked upon one another).

<code>zcol</code>	vector of integers indicating which columns of <code>pred</code> will be used in $Z_i$ . That is, <code>pred[,zcol]</code> is the $Z_i$ matrices (stacked upon one another).
<code>prior</code>	A list with four components specifying the hyperparameters of the prior distribution applied to <code>sigma2</code> and <code>psi</code> . The components must be named "a", "b", "c", and "Dinv". All are scalars except for "Dinv", which is a matrix of dimension $c(\text{length}(zcol), \text{length}(zcol))$ .
<code>vmax</code>	optional matrix of dimension $c(\max(\text{occ}), \max(\text{occ}))$ from which the $V_i$ matrices will be extracted. In a longitudinal dataset, <code>vmax</code> would represent the $V_i$ matrix for an individual with responses at all possible occasions $1, 2, \dots, n_{\max} = \max(\text{occ})$ ; for individuals with responses at only a subset of these occasions, the $V_i$ will be obtained by extracting the rows and columns of <code>vmax</code> for those occasions. If no <code>vmax</code> is specified by the user, an identity matrix is used. In most applications of this model one will want to have $V_i = \text{identity}$ , so most of the time this argument can be omitted.
<code>occ</code>	vector of same length as <code>y</code> indicating the "occasions" for the elements of <code>y</code> . This argument is relevant only if a non-identity <code>vmax</code> is specified. In a longitudinal dataset where each individual is measured on at most <code>nmax</code> distinct occasions, each element of <code>y</code> corresponds to one subject-occasion, and the elements of <code>occ</code> should be coded as $1, 2, \dots, n_{\max}$ to indicate these occasion labels. (You should label the occasions as $1, 2, \dots, n_{\max}$ even if they are not equally spaced in time; the actual times of measurement will be incorporated into the matrix "pred".)
<code>start</code>	optional starting values of the parameters. If this argument is not given then the function chooses its own starting values. This argument should be a list of three elements named "beta", "psi", and "sigma2". Note that "beta" should be a vector of the same length as "xcol", "psi" should be a matrix of dimension $c(\text{length}(zcol), \text{length}(zcol))$ , and "sigma2" should be a scalar.
<code>maxits</code>	maximum number of cycles to be performed. The algorithm runs to convergence or until "maxits" iterations, whichever comes first.
<code>eps</code>	convergence criterion. The algorithm is considered to have converged if the relative differences in all parameters from one iteration to the next are less than <code>eps</code> —that is, if $\text{all}(\text{abs}(\text{new-old}) < \text{eps} * \text{abs}(\text{old}))$ .

## Details

The algorithm is described in the appendix of Schafer (1998). Scoring is carried out on  $\log(\text{sigma2})$  and the nonredundant elements of the inverse of `psi/sigma2`, taking logs of the diagonal elements. Upon convergence, the estimates represent the mode of the joint posterior density of  $1/\text{sigma2}$  and the inverse of `psi`, marginalized (i.e. integrated) over `beta`.

The prior distribution applied to the within-unit residual variance is scaled inverted-chisquare,

$$\text{sigma2} \sim a / \text{chisq}(b),$$

where  $\text{chisq}(b)$  denotes a chisquare random variable with  $b$  degrees of freedom, and  $a$  and  $b$  are user-defined hyperparameters. Values for the hyperparameters may be chosen by regarding  $a/b$  as a rough prior guess for `sigma2`, and as the imaginary degrees of freedom on which this guess is based.

The prior distribution applied to the between-unit covariance matrix is inverted Wishart,

$$\text{psiinv} \sim W(c, D),$$

where  $\psi^{-1}$  is the inverse of the between-unit covariance matrix  $\psi$ , and  $W(c,D)$  denotes a Wishart distribution with degrees of freedom  $c$  and scale matrix  $D$ . Values for the hyperparameters may be chosen by regarding  $D^{-1}/c$  (the inverse of  $D$  divided by  $c$ ) as a rough prior guess for  $\psi$ , and  $c$  as the imaginary degrees of freedom on which this guess is based.

An improper uniform prior density function is applied to the fixed effects  $\beta$ .

## Value

a list containing the following components.

<code>beta</code>	vector of same length as "xcol" containing estimated fixed effects. This estimate represents the posterior mean for $\beta$ , conditional upon the estimated values of the variance parameters <code>sigma2</code> and <code>psi</code> .
<code>sigma2</code>	estimate of residual error variance.
<code>psi</code>	matrix of dimension $c(\text{length}(zcol),\text{length}(zcol))$ containing estimated variances and covariances of the random effects.
<code>converged</code>	T if the algorithm converged, F if it did not.
<code>iter</code>	number of iterations actually performed. Will be equal to "maxits" if <code>converged=F</code> .
<code>reject</code>	a logical vector of length <code>iter</code> indicating, for each iteration, whether the scoring estimates were rejected and replaced by ECME estimates (T), or whether the scoring estimates were accepted (F). Scoring estimates are rejected if they do not increase the log-posterior density.
<code>logpost</code>	vector of length "iter" reporting the value of the log-posterior density at each iteration.
<code>cov.beta</code>	matrix of dimension $c(\text{length}(xcol),\text{length}(xcol))$ containing estimated variances and covariances for elements of "beta". These are conventional estimates which regard the variance parameters ( <code>sigma2</code> and <code>psi</code> ) as fixed at their estimated values.

## References

Schafer, J.L. (1998) Some improved procedures for linear mixed models. Submitted to Journal of the American Statistical Association.

## See Also

[ecmml.lmm](#), [ecmerml.lmm](#), [fastml.lmm](#), [fastrml.lmm](#), [mgibbs.lmm](#), [fastmcmc.lmm](#), [example.lmm](#)

## Examples

```
## Not run:
For a detailed example, see the file "example.lmm.R" distributed
with this library.

## End(Not run)
```

---

fastrml.lmm	<i>Rapidly converging algorithm for restricted maximum-likelihood (RML) estimation in linear mixed models</i>
-------------	---

---

### Description

Computes RML estimates of parameters in linear mixed models using the rapidly converging procedure described by Schafer (1998), which combines Fisher scoring with an ECME algorithm.

For a description of the model, see the "Details" section below.

### Usage

```
fastrml.lmm(y, subj, pred, xcol, zcol, vmax, occ, start,
            maxits=50, eps=0.0001)
```

### Arguments

y	vector of responses. This is simply the individual $y_i$ vectors stacked upon one another. Each element of y represents the observed response for a particular subject-occasion, or for a particular unit within a cluster.
subj	vector of same length as y, giving the subject (or cluster) indicators i for the elements of y. For example, suppose that y is $c(y_1, y_2, y_3, y_4)$ where $\text{length}(y_1)=2$ , $\text{length}(y_2)=3$ , $\text{length}(y_3)=2$ , and $\text{length}(y_4)=7$ . Then subj should be $c(1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 4, 4, 4, 4)$ .
pred	matrix of covariates used to predict y. The number of rows should be $\text{length}(y)$ . The first column will typically be constant (one), and the remaining columns correspond to other variables appearing in $X_i$ and $Z_i$ .
xcol	vector of integers indicating which columns of pred will be used in $X_i$ . That is, $\text{pred}[,xcol]$ is the $X_i$ matrices (stacked upon one another).
zcol	vector of integers indicating which columns of pred will be used in $Z_i$ . That is, $\text{pred}[,zcol]$ is the $Z_i$ matrices (stacked upon one another).
vmax	optional matrix of dimension $c(\max(\text{occ}), \max(\text{occ}))$ from which the $V_i$ matrices will be extracted. In a longitudinal dataset, vmax would represent the $V_i$ matrix for an individual with responses at all possible occasions $1, 2, \dots, n_{\max}=\max(\text{occ})$ ; for individuals with responses at only a subset of these occasions, the $V_i$ will be obtained by extracting the rows and columns of vmax for those occasions. If no vmax is specified by the user, an identity matrix is used. In most applications of this model one will want to have $V_i = \text{identity}$ , so most of the time this argument can be omitted.
occ	vector of same length as y indicating the "occasions" for the elements of y. This argument is relevant only if a non-identity vmax is specified. In a longitudinal dataset where each individual is measured on at most $n_{\max}$ distinct occasions, each element of y corresponds to one subject-occasion, and the elements of occ should be coded as $1, 2, \dots, n_{\max}$ to indicate these occasion labels. (You should label the occasions as $1, 2, \dots, n_{\max}$ even if they are not equally spaced in time; the actual times of measurement will be incorporated into the matrix "pred".)

start	optional starting values of the parameters. If this argument is not given then the function chooses its own starting values. This argument should be a list of three elements named "beta", "psi", and "sigma2". Note that "beta" should be a vector of the same length as "xcol", "psi" should be a matrix of dimension $c(\text{length}(zcol), \text{length}(zcol))$ , and "sigma2" should be a scalar.
maxits	maximum number of cycles to be performed. The algorithm runs to convergence or until "maxits" iterations, whichever comes first.
eps	convergence criterion. The algorithm is considered to have converged if the relative differences in all parameters from one iteration to the next are less than eps—that is, if $\text{all}(\text{abs}(\text{new}-\text{old}) < \text{eps} * \text{abs}(\text{old}))$ .

### Details

A full description of the algorithm is given in Section 3 of Schafer (1998). Scoring is carried out on  $\log(\text{sigma2})$  and the nonredundant elements of the inverse of  $\text{psi}/\text{sigma2}$ , taking logs of the diagonal elements. Improved estimates of variances and covariances are described in Section 4.

The model, which is typically applied to longitudinal or clustered responses, is

$$y_i = X_i \beta + Z_i b_i + e_i, \quad i=1, \dots, m,$$

where

$y_i = (n_i \times 1)$  response vector for subject or cluster  $i$ ;  $X_i = (n_i \times p)$  matrix of covariates;  $Z_i = (n_i \times q)$  matrix of covariates;  $\beta = (p \times 1)$  vector of coefficients common to the population (fixed effects);  $b_i = (q \times 1)$  vector of coefficients specific to subject or cluster  $i$  (random effects); and  $e_i = (n_i \times 1)$  vector of residual errors.

The vector  $b_i$  is assumed to be normally distributed with mean zero and unstructured covariance matrix  $\text{psi}$ ,

$$b_i \sim N(0, \text{psi}) \text{ independently for } i=1, \dots, m.$$

The residual vector  $e_i$  is assumed to be

$$e_i \sim N(0, \text{sigma2} * V_i)$$

where  $V_i$  is a known  $(n_i \times n_i)$  matrix. In most applications,  $V_i$  is the identity matrix.

### Value

a list containing the following components.

beta	vector of same length as "xcol" containing estimated fixed effects.
sigma2	estimate of residual error variance.
psi	matrix of dimension $c(\text{length}(zcol), \text{length}(zcol))$ containing estimated variances and covariances of the random effects.
converged	T if the algorithm converged, F if it did not.
iter	number of iterations actually performed. Will be equal to "maxits" if converged=F.
reject	a logical vector of length iter indicating, for each iteration, whether the scoring estimates were rejected and replaced by ECME estimates (T), or whether the scoring estimates were accepted (F). Scoring estimates are rejected if they do not increase the loglikelihood.

<code>loglik</code>	vector of length "iter" reporting the value of the loglikelihood at each iteration.
<code>cov.beta</code>	matrix of dimension $c(\text{length}(xcol), \text{length}(xcol))$ containing estimated variances and covariances for elements of "beta". These are conventional estimates which regard the variance parameters ( $\sigma^2$ and $\psi$ ) as fixed at their RML estimates.
<code>b.hat</code>	a matrix with $\text{length}(zcol)$ rows and $m$ columns, where <code>b.hat[,i]</code> is an empirical Bayes estimate of $b_i$ .
<code>cov.b</code>	an array of dimension $\text{length}(zcol)$ by $\text{length}(zcol)$ by $m$ , where <code>cov.b[:,i]</code> is an empirical Bayes estimate of the covariance matrix associated with $b_i$ . These are conventional estimates which regard the variance parameters ( $\sigma^2$ and $\psi$ ) as fixed at their RML estimates.
<code>cov.beta.new</code>	matrix of dimension $c(\text{length}(xcol), \text{length}(xcol))$ containing estimated variances and covariances for elements of "beta". These are improved estimates which account for uncertainty in estimating the variance parameters ( $\sigma^2$ and $\psi$ ).
<code>cov.b.new</code>	an array of dimension $\text{length}(zcol)$ by $\text{length}(zcol)$ by $m$ , where <code>cov.b.new[:,i]</code> is an estimated covariance matrix for $b_i$ . These are improved estimates which account for uncertainty in estimating the variance parameters ( $\sigma^2$ and $\psi$ ).
<code>cov.b.beta.new</code>	an array of dimension $\text{length}(zcol)$ by $\text{length}(xcol)$ by $m$ , where <code>cov.b.beta.new[:,i]</code> contains the estimated covariances between $b_i$ and beta. These are improved estimates which account for uncertainty in estimating the variance parameters ( $\sigma^2$ and $\psi$ ). Note that conventional estimates which regard $\sigma^2$ and $\psi$ as fixed assume zero covariance between each $b_i$ and beta.

## References

Schafer, J.L. (1998) Some improved procedures for linear mixed models. Submitted to Journal of the American Statistical Association.

## See Also

[ecmestl.lmm](#), [ecmerml.lmm](#), [fastml.lmm](#), [fastmode.lmm](#), [mgibbs.lmm](#), [fastmcmc.lmm](#), [example.lmm](#)

## Examples

```
## Not run:
For a detailed example, see the file "example.lmm.R" distributed
with this library.

## End(Not run)
```

mgibbs.lmm

*Modified Gibbs sampler for Bayesian inference in linear mixed models***Description**

Simulates posterior draws of parameters in linear mixed models using a Markov chain Monte Carlo (MCMC) procedure, the modified Gibbs sampler described by Schafer (1998). This algorithm may be slow, requiring a large number of cycles to achieve stationarity. In most cases, "fastmcmc.lmm" will perform better. This algorithm is provided mainly for comparison against "fastmcmc.lmm".

For a description of the model and the prior distribution, see the "Details" section below.

**Usage**

```
mgibbs.lmm(y, subj, pred, xcol, zcol, prior, seed, vmax, occ,
           start, iter=1000)
```

**Arguments**

y	vector of responses. This is simply the individual $y_i$ vectors stacked upon one another. Each element of y represents the observed response for a particular subject-occasion, or for a particular unit within a cluster.
subj	vector of same length as y, giving the subject (or cluster) indicators $i$ for the elements of y. For example, suppose that y is $c(y_1, y_2, y_3, y_4)$ where $\text{length}(y_1)=2$ , $\text{length}(y_2)=3$ , $\text{length}(y_3)=2$ , and $\text{length}(y_4)=7$ . Then subj should be $c(1, 1, 2, 2, 2, 3, 3, 4, 4, 4, 4, 4, 4)$ .
pred	matrix of covariates used to predict y. The number of rows should be $\text{length}(y)$ . The first column will typically be constant (one), and the remaining columns correspond to other variables appearing in $X_i$ and $Z_i$ .
xcol	vector of integers indicating which columns of pred will be used in $X_i$ . That is, $\text{pred}[,xcol]$ is the $X_i$ matrices (stacked upon one another).
zcol	vector of integers indicating which columns of pred will be used in $Z_i$ . That is, $\text{pred}[,zcol]$ is the $Z_i$ matrices (stacked upon one another).
prior	A list with four components specifying the hyperparameters of the prior distribution applied to $\sigma^2$ and $\psi$ . The components must be named "a", "b", "c", and "Dinv". All are scalars except for "Dinv", which is a matrix of dimension $c(\text{length}(zcol), \text{length}(zcol))$ .
seed	Seed for random number generator. This should be a positive integer.
vmax	optional matrix of dimension $c(\max(\text{occ}), \max(\text{occ}))$ from which the $V_i$ matrices will be extracted. In a longitudinal dataset, vmax would represent the $V_i$ matrix for an individual with responses at all possible occasions $1, 2, \dots, n_{\max} = \max(\text{occ})$ ; for individuals with responses at only a subset of these occasions, the $V_i$ will be obtained by extracting the rows and columns of vmax for those occasions. If no vmax is specified by the user, an identity matrix is used. In most applications of this model one will want to have $V_i = \text{identity}$ , so most of the time this argument can be omitted.

<code>occ</code>	vector of same length as <code>y</code> indicating the "occasions" for the elements of <code>y</code> . This argument is relevant only if a non-identity <code>vmax</code> is specified. In a longitudinal dataset where each individual is measured on at most <code>nmax</code> distinct occasions, each element of <code>y</code> corresponds to one subject-occasion, and the elements of <code>occ</code> should be coded as 1,2,...,nmax to indicate these occasion labels. (You should label the occasions as 1,2,...,nmax even if they are not equally spaced in time; the actual times of measurement will be incorporated into the matrix "pred".)
<code>start</code>	optional starting values of the parameters. If this argument is not given then the function chooses its own starting values. This argument should be a list of three elements named "beta", "psi", and "sigma2". Note that "beta" should be a vector of the same length as "xcol", "psi" should be a matrix of dimension $c(\text{length}(zcol), \text{length}(zcol))$ , and "sigma2" should be a scalar.
<code>iter</code>	number of cycles of the modified Gibbs sampler to be performed.

## Details

The algorithm is described in Section 5 of Schafer (1998).

The model, which is typically applied to longitudinal or clustered responses, is

$$y_i = X_i \beta + Z_i b_i + e_i, \quad i=1, \dots, m,$$

where

$y_i$  = ( $n_i \times 1$ ) response vector for subject or cluster  $i$ ;  $X_i$  = ( $n_i \times p$ ) matrix of covariates;  $Z_i$  = ( $n_i \times q$ ) matrix of covariates;  $\beta$  = ( $p \times 1$ ) vector of coefficients common to the population (fixed effects);  $b_i$  = ( $q \times 1$ ) vector of coefficients specific to subject or cluster  $i$  (random effects); and  $e_i$  = ( $n_i \times 1$ ) vector of residual errors.

The vector  $b_i$  is assumed to be normally distributed with mean zero and unstructured covariance matrix  $\psi_i$ ,

$$b_i \sim N(0, \psi_i) \text{ independently for } i=1, \dots, m.$$

The residual vector  $e_i$  is assumed to be

$$e_i \sim N(0, \sigma^2 V_i)$$

where  $V_i$  is a known ( $n_i \times n_i$ ) matrix. In most applications,  $V_i$  is the identity matrix.

The prior distribution applied to the within-unit residual variance is scaled inverted-chisquare,

$$\sigma^2 \sim a / \text{chisq}(b),$$

where  $\text{chisq}(b)$  denotes a chisquare random variable with  $b$  degrees of freedom, and  $a$  and  $b$  are user-defined hyperparameters. Values for the hyperparameters may be chosen by regarding  $a/b$  as a rough prior guess for  $\sigma^2$ , and as the imaginary degrees of freedom on which this guess is based.

The prior distribution applied to the between-unit covariance matrix is inverted Wishart,

$$\psi_i \sim W(c, D),$$

where  $\psi_i$  is the inverse of the between-unit covariance matrix  $\psi_i$ , and  $W(c, D)$  denotes a Wishart distribution with degrees of freedom  $c$  and scale matrix  $D$ . Values for the hyperparameters may be chosen by regarding  $D_{inv}/c$  (the inverse of  $D$  divided by  $c$ ) as a rough prior guess for  $\psi_i$ , and  $c$  as the imaginary degrees of freedom on which this guess is based.

An improper uniform prior density function is applied to the fixed effects  $\beta$ .

**Value**

a list containing the following components.

<code>beta</code>	simulated value of coefficients <code>beta</code> after "iter" cycles of the modified Gibbs sampler. This is a vector of the same length as <code>xcol</code> .
<code>sigma2</code>	simulated value of the residual variance <code>sigma2</code> after "iter" cycles of the modified Gibbs sampler.
<code>psi</code>	simulated value of the between-unit covariance matrix <code>psi</code> after "iter" cycles of the modified Gibbs sampler.
<code>sigma2.series</code>	vector of length "iter" containing the entire history of simulated values of <code>sigma2</code> . That is, <code>sigma2.series[t]</code> contains the value of <code>sigma2</code> at cycle <code>t</code> .
<code>psi.series</code>	array of dimension <code>c(length(zcol),length(zcol),iter)</code> containing the entire history of simulated values of <code>psi</code> . That is, <code>psi.series[,t]</code> contains the value of <code>psi</code> at cycle <code>t</code> .

**References**

Schafer, J.L. (1998) Some improved procedures for linear mixed models. Submitted to Journal of the American Statistical Association.

**See Also**

[ecmest.lmm](#), [ecmerml.lmm](#), [fastml.lmm](#), [fastrml.lmm](#), [fastmode.lmm](#), [fastmcmc.lmm](#), [example.lmm](#)

**Examples**

```
## Not run:  
For a detailed example, see the file "example.lmm.R" distributed  
with this library.  
  
## End(Not run)
```

# Index

## \*Topic **models**

- ecmeml.lmm, 1
- ecmerml.lmm, 4
- example.lmm, 7
- fastmcmc.lmm, 10
- fastml.lmm, 13
- fastmode.lmm, 16
- fastrml.lmm, 18
- mgibbs.lmm, 21

ecmeml.lmm, 1, 6, 7, 13, 15, 18, 21, 24  
ecmerml.lmm, 4, 4, 7, 13, 15, 18, 21, 24  
example.lmm, 4, 6, 7, 13, 15, 18, 21, 24

fastmcmc.lmm, 4, 6, 7, 10, 15, 18, 21, 24  
fastml.lmm, 4, 6, 7, 13, 13, 18, 21, 24  
fastmode.lmm, 4, 6, 7, 13, 15, 16, 21, 24  
fastrml.lmm, 4, 6, 7, 13, 15, 18, 18, 24

mgibbs.lmm, 4, 6, 7, 13, 15, 18, 21, 21