Package 'lmm'

July 22, 2025

Version 1.4

Date 2023-8-19

Title Linear Mixed Models

Author Original by Joseph L. Schafer

Maintainer Jing hua Zhao <jinghuazhao@hotmail.com>

Depends R (>= 2.0.0)

LazyData Yes

LazyLoad Yes

Description It implements Expectation/Conditional Maximization Either (ECME)
and rapidly converging algorithms as well as
Bayesian inference for linear mixed models,
which is described in Schafer, J.L. (1998)
``Some improved procedures for linear mixed models".
Dept. of Statistics, The Pennsylvania State University.

License Unlimited

URL https://github.com/jinghuazhao/R

NeedsCompilation yes

Repository CRAN

Date/Publication 2023-08-19 19:52:32 UTC

Contents

ecmeml *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" will perform better. This function is provided mainly for comparison against "fastml".

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

Usage

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

Arguments

y	vector of responses. This is simply the individual yi 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(y1,y2,y3,y4) where length(y1)=2, length(y2)=3, length(y3)=2, and length(y4)=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 length(y). The first column will typically be constant (one), and the remaining columns correspond to other variables appearing in Xi and Zi.
xcol	vector of integers indicating which columns of pred will be used in Xi. That is, pred[,xcol] is the Xi matrices (stacked upon one another.
zcol	vector of integers indicating which columns of pred will be used in Zi. That is, pred[,zcol] is the Zi matrices (stacked upon one another).
vmax	optional matrix of dimension c(max(occ),max(occ)) from which the Vi matrices will be extracted. In a longitudinal dataset, vmax would represent the Vi matrix for an individual with responses at all possible occasions 1,2,. . . ,nmax=max(occ); for individuals with responses at only a subset of these occasions, the Vi 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 Vi = 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 nmax distinct occasions, each element of y corresponds to one subject-occasion, and the elements of occ 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".)

`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(length(zcol),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 all(abs(new-old)<eps*abs(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

yi = Xi%*%beta + Zi%*%bi + ei , i=1,...,m,

where

yi = (ni x 1) response vector for subject or cluster i; Xi = (ni x p) matrix of covariates; Zi = (ni x q) matrix of covariates; beta = (p x 1) vector of coefficients common to the population (fixed effects); bi = (q x 1) vector of coefficients specific to subject or cluster i (random effects); and ei = (ni x 1) vector of residual errors.

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

bi $\sim$ N(0,psi) independently for i=1,...,m.

The residual vector ei is assumed to be

ei $\sim$ N(0,sigma2*Vi)

where Vi is a known (ni x ni) matrix. In most applications, Vi 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(length(zcol),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.

cov.beta	matrix of dimension c(length(xcol),length(xcol)) containing estimated variances and covariances for elements of "beta". These are conventional estimates which regard the variance parameters (sigma2 and psi) as fixed at their ML estimates.
b.hat	a matrix with length(zcol) rows and m columns, where b.hat[,i] is an empirical Bayes estimate of bi.
cov.b	an array of dimension length(zcol) by length(zcol) by m, where cov.b[,,i] is an empirical Bayes estimate of the covariance matrix associated with bi. These are conventional estimates which regard the variance parameters (sigma2 and psi) as fixed at their ML estimates. (An improved version which incorporates variance-parameter uncertainty is available from the function "fastrml".)

References

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

See Also

ecmerml, fastml, fastrml, fastmode, mgibbs, fastmcmc, example

Examples

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

## End(Not run)
```

---

ecmerml — *ECME algorithm for restricted maximum-likelihood (RML) estimation in linear mixed models*

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" will perform better. This function is provided mainly for comparison against "fastrml".

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

Usage

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

Arguments

y	vector of responses. This is simply the individual yi 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(y1,y2,y3,y4) where length(y1)=2, length(y2)=3, length(y3)=2, and length(y4)=7. Then subj should be c(1,1,2,2,2,3,3,4,4,4,4,4,4,4,4).
pred	matrix of covariates used to predict y. The number of rows should be length(y). The first column will typically be constant (one), and the remaining columns correspond to other variables appearing in Xi and Zi.
xcol	vector of integers indicating which columns of pred will be used in Xi. That is, pred[,xcol] is the Xi matrices (stacked upon one another.
zcol	vector of integers indicating which columns of pred will be used in Zi. That is, pred[,zcol] is the Zi matrices (stacked upon one another).
vmax	optional matrix of dimension c(max(occ),max(occ)) from which the Vi matrices will be extracted. In a longitudinal dataset, vmax would represent the Vi matrix for an individual with responses at all possible occasions 1,2,. . . ,nmax=max(occ); for individuals with responses at only a subset of these occasions, the Vi 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 Vi = 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 nmax distinct occasions, each element of y corresponds to one subject-occasion, and the elements of occ 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".)
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(length(zcol),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 all(abs(new-old)<eps*abs(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

yi = Xi%*%beta + Zi%*%bi + ei , i=1,. . . ,m,

where

yi = (ni x 1) response vector for subject or cluster i; Xi = (ni x p) matrix of covariates; Zi = (ni x q) matrix of covariates; beta = (p x 1) vector of coefficients common to the population (fixed effects); bi = (q x 1) vector of coefficients specific to subject or cluster i (random effects); and ei = (ni x 1) vector of residual errors.

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

bi $\sim$ N(0,psi) independently for i=1,. . . ,m.

The residual vector ei is assumed to be

ei $\sim$ N(0,sigma2*Vi)

where Vi is a known (ni x ni) matrix. In most applications, Vi 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(length(zcol),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 "restricted" loglikelihood at each iteration.
`cov.beta`	matrix of dimension c(length(xcol),length(xcol)) containing estimated variances and covariances for elements of "beta". These are conventional estimates which regard the variance parameters (sigma2 and psi) as fixed at their RML estimates.
`b.hat`	a matrix with length(zcol) rows and m columns, where b.hat[,i] is an empirical Bayes estimate of bi.
`cov.b`	an array of dimension length(zcol) by length(zcol) by m, where cov.b[,,i] is an empirical Bayes estimate of the covariance matrix associated with bi. These are conventional estimates which regard the variance parameters (sigma2 and psi) as fixed at their RML estimates. (An improved version which incorporates variance-parameter uncertainty is available from the function "fastrml".)

References

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

See Also

ecmeml, fastml, fastrml, fastmode, mgibbs, fastmcmc, example

Examples

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


## End(Not run)
```

---

example *lmm package example command file*

Description

The data as contained in marijuana is used to fit a compound symmetry model with a fixed effect for each occasion and a random intercept for each subject.

Since the six measurements per subject were not clearly ordered in time, instead of a model with time of measurement entered with linear (or perhaps higher-order polynomial) effects, the model has an intercept and five dummy codes to allow the population means for the six occasions to be estimated freely. For a subject i with no missing values, the covariate matrices will be

Xi=(1 1 0 0 0 0, 1 0 1 0 0 0, 1 0 0 1 0 0, 1 0 0 0 1 0, 1 0 0 0 0 1) and Zi=(1,1,1,1,1,1)

The Xi's and Zi's are combined into a single matrix called pred (Zi is merely the first column of Xi), simply the matrices Xi (i=1,...,9), stacked upon each other.

See Also

ecmeml, ecmerml, fastml, fastrml, fastmcmc, fastmode, mgibbs

Examples

```
### Model specification ###
data(marijuana)
# To work only on those with complete data
marijuana <- subset(marijuana,!is.na(y))
attach(marijuana)
pred <- cbind(int,dummy1,dummy2,dummy3,dummy4,dummy5)
xcol <- 1:6
zcol <- 1

### ML Estimation ###
ecmeml.result <- ecmeml(y,subj,pred,xcol,zcol)
fastml.result <- fastml(y,subj,pred,xcol,zcol)
#
# which converged in 212 and 8 cycles, respectively. For example, the
# first elemenent of the ML estimate of the fixed effects (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.
#
```

```
beta.hat <- fastml.result$beta
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])

### RML estimation ###
ecmerml.result <- ecmerml(y,subj,pred,xcol,zcol)
fastrml.result <- fastrml(y,subj,pred,xcol,zcol)

### Improved variance estimation in Section 4 ###
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)
#
# which reproduces Table 2 and compares 95% interval estimates
# under the new method to conventional empirical Bayes intervals.

### MCMC in Section 5 ###
prior <- list(a=3*100,b=3,c=3,Dinv=3*5)
gibbs.result <- mgibbs(y,subj,pred,xcol,zcol,prior=prior,seed=1234,iter=5000)
fmcmc.result <- fastmcmc(y,subj,pred,xcol,zcol,prior=prior,seed=2345,iter=5000)
#
# which run 5,000 cycles for each algorithm and generates Figure 1.
#
# library(ts)
par(mfrow=c(2,1))
acf(log(gibbs.result$psi.series[1,1,]),lag.max=10, ylim=0:1)
acf(log(fmcmc.result$psi.series[1,1,]),lag.max=10, ylim=0:1)
detach(marijuana)
```

`fastmcmc` *Rapidly converging Markov chain Monte Carlo algorithm for Bayesian inference in linear mixed models*

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". The results from a call to "fastmode" are returned along with the MCMC results.

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

Usage

```
fastmcmc(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 yi 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(y1,y2,y3,y4) where length(y1)=2, length(y2)=3, length(y3)=2, and length(y4)=7. Then subj should be c(1,1,2,2,2,3,3,4,4,4,4,4,4,4,4).
pred	matrix of covariates used to predict y. The number of rows should be length(y). The first column will typically be constant (one), and the remaining columns correspond to other variables appearing in Xi and Zi.
xcol	vector of integers indicating which columns of pred will be used in Xi. That is, pred[,xcol] is the Xi matrices (stacked upon one another.
zcol	vector of integers indicating which columns of pred will be used in Zi. That is, pred[,zcol] is the Zi matrices (stacked upon one another).
prior	A list with four components specifying the hyperparameters of the prior distribution applied to sigma2 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(length(zcol),length(zcol)).
seed	Seed for random number generator. This should be a positive integer.
vmax	optional matrix of dimension c(max(occ),max(occ)) from which the Vi matrices will be extracted. In a longitudinal dataset, vmax would represent the Vi matrix for an individual with responses at all possible occasions 1,2,...,nmax=max(occ); for individuals with responses at only a subset of these occasions, the Vi 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 Vi = 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 nmax distinct occasions, each element of y corresponds to one subject-occasion, and the elements of occ 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".)
start.mode	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 c(length(zcol),length(zcol)), and "sigma2" should be a scalar.

maxits	maximum number of cycles of the mode-finding procedure. The algorithm runs to convergence or until "maxits" iterations, whichever comes first.
eps	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 eps–that is, if all(abs(new-old)<eps*abs(old)).
iter	number of cycles of the MCMC procedure to be performed.
start.mcmc	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.
df	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

yi = Xi%*%beta + Zi%*%bi + ei , i=1,. . . ,m,

where

yi = (ni x 1) response vector for subject or cluster i; Xi = (ni x p) matrix of covariates; Zi = (ni x q) matrix of covariates; beta = (p x 1) vector of coefficients common to the population (fixed effects); bi = (q x 1) vector of coefficients specific to subject or cluster i (random effects); and ei = (ni x 1) vector of residual errors.

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

bi $\sim$ N(0,psi) independently for i=1,. . . ,m.

The residual vector ei is assumed to be

ei $\sim$ N(0,sigma2*Vi)

where Vi is a known (ni x ni) matrix. In most applications, Vi is the identity matrix.

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

sigma2 $\sim$ a / chisq(b),

where chisq(b) denotes a chisquare random variable with b degrees of freedom, and a and b are user-defined hyperparameters. Values for the hyperparmeters 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,

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

where psiinv 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 Dinv/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.

`beta`	simulated value of coefficients beta after "iter" cycles of the MCMC algorithm. This is a vector of the same length as xcol.
`sigma2`	simulated value of the residual variance sigma2 after "iter" cycles of the MCMC algorithm.
`psi`	simulated value of the between-unit covariance matrix psi after "iter" cycles of the MCMC algorithm.
`sigma2.series`	vector of length "iter" containing the entire history of simulated values of sigma2. That is, sigma2.series[t] contains the value of sigma2 at cycle t.
`psi.series`	array of dimension c(length(zcol),length(zcol),iter) containing the entire history of simulated values of psi. That is, psi.series[,,t] contains the value of psi at cycle t.
`ratios`	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.
`reject`	logical vector of length "iter" indicating, for each cycle of the algorithm, whether the Metropolis-Hastings candidate was accepted (T) or rejected (F).
`mode.list`	a list containing the results of the mode-finding procedure. The contents of this list are identical to those produced by "fastmode". For more information, see the help file for "fastmode".

References

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

See Also

`ecmeml`, `ecmerml`, `fastml`, `fastrml`, `fastmode`, `mgibbs`, `example`

Examples

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


## End(Not run)
```

`fastml` *Rapidly converging algorithm for maximum-likelihood (ML) estimation in linear mixed models*

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

Arguments

`y`	vector of responses. This is simply the individual yi 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(y1,y2,y3,y4) where length(y1)=2, length(y2)=3, length(y3)=2, and length(y4)=7. Then subj should be c(1,1,2,2,2,3,3,4,4,4,4,4,4,4,4).
`pred`	matrix of covariates used to predict y. The number of rows should be length(y). The first column will typically be constant (one), and the remaining columns correspond to other variables appearing in Xi and Zi.
`xcol`	vector of integers indicating which columns of pred will be used in Xi. That is, pred[,xcol] is the Xi matrices (stacked upon one another.
`zcol`	vector of integers indicating which columns of pred will be used in Zi. That is, pred[,zcol] is the Zi matrices (stacked upon one another).
`vmax`	optional matrix of dimension c(max(occ),max(occ)) from which the Vi matrices will be extracted. In a longitudinal dataset, vmax would represent the Vi matrix for an individual with responses at all possible occasions 1,2,...,nmax=max(occ); for individuals with responses at only a subset of these occasions, the Vi 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 Vi = 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 nmax distinct occasions, each element of y corresponds to one subject-occasion, and the elements of occ 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".)

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(length(zcol),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 all(abs(new-old)<eps*abs(old)).

Details

A full description of the algorithm is given in Section 3 of Schafer (1998). Scoring is carried out on log(sigma2) and the nonredundant elements of the inverse of psi/sigma2, taking logs of the diagonal elements.

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

yi = Xi%*%beta + Zi%*%bi + ei , i=1,. . . ,m,

where

yi = (ni x 1) response vector for subject or cluster i; Xi = (ni x p) matrix of covariates; Zi = (ni x q) matrix of covariates; beta = (p x 1) vector of coefficients common to the population (fixed effects); bi = (q x 1) vector of coefficients specific to subject or cluster i (random effects); and ei = (ni x 1) vector of residual errors.

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

bi $\sim$ N(0,psi) independently for i=1,. . . ,m.

The residual vector ei is assumed to be

ei $\sim$ N(0,sigma2*Vi)

where Vi is a known (ni x ni) matrix. In most applications, Vi 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(length(zcol),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.

loglik	vector of length "iter" reporting the value of the loglikelihood at each iteration.
cov.beta	matrix of dimension c(length(xcol),length(xcol)) containing estimated variances and covariances for elements of "beta". These are conventional estimates which regard the variance parameters (sigma2 and psi) as fixed at their ML estimates.
b.hat	a matrix with length(zcol) rows and m columns, where b.hat[,i] is an empirical Bayes estimate of bi.
cov.b	an array of dimension length(zcol) by length(zcol) by m, where cov.b[,,i] is an empirical Bayes estimate of the covariance matrix associated with bi. These are conventional estimates which regard the variance parameters (sigma2 and psi) as fixed at their ML estimates. (An improved version which incorporates variance-parameter uncertainty is available from the function "fastrml".)

References

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

See Also

ecmeml, ecmerml, fastrml, fastmode, mgibbs, fastmcmc, example

Examples

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

## End(Not run)
```

---

fastmode — *Rapidly converging algorithm for calculating posterior modes in linear mixed models*

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". The model is identical to that of "fastrml" with the addition of prior distributions for the variance parameters.

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

Usage

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

Arguments

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

y	vector of responses. This is simply the individual yi 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(y1,y2,y3,y4) where length(y1)=2, length(y2)=3, length(y3)=2, and length(y4)=7. Then subj should be c(1,1,2,2,2,3,3,4,4,4,4,4,4,4,4).
pred	matrix of covariates used to predict y. The number of rows should be length(y). The first column will typically be constant (one), and the remaining columns correspond to other variables appearing in Xi and Zi.
xcol	vector of integers indicating which columns of pred will be used in Xi. That is, pred[,xcol] is the Xi matrices (stacked upon one another.
zcol	vector of integers indicating which columns of pred will be used in Zi. That is, pred[,zcol] is the Zi matrices (stacked upon one another).
prior	A list with four components specifying the hyperparameters of the prior distribution applied to sigma2 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(length(zcol),length(zcol)).
vmax	optional matrix of dimension c(max(occ),max(occ)) from which the Vi matrices will be extracted. In a longitudinal dataset, vmax would represent the Vi matrix for an individual with responses at all possible occasions 1,2,...,nmax=max(occ); for individuals with responses at only a subset of these occasions, the Vi 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 Vi = 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 nmax distinct occasions, each element of y corresponds to one subject-occasion, and the elements of occ 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".)
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(length(zcol),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 all(abs(new-old)<eps*abs(old)).

Details

The algorithm is described in the appendix of Schafer (1998). Scoring is carried out on log(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/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,

sigma2 $\sim$ a / chisq(b),

where chisq(b) denotes a chisquare random variable with b degrees of freedom, and a and b are user-defined hyperparameters. Values for the hyperparmeters 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,

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

where psiinv 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 Dinv/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.

`beta`	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 sigma2 and psi.
`sigma2`	estimate of residual error variance.
`psi`	matrix of dimension c(length(zcol),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 log-posterior density.
`logpost`	vector of length "iter" reporting the value of the log-posterior density at each iteration.
`cov.beta`	matrix of dimension c(length(xcol),length(xcol)) containing estimated variances and covariances for elements of "beta". These are conventional estimates which regard the variance parameters (sigma2 and psi) 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

ecmeml, ecmerml, fastml, fastrml, mgibbs, fastmcmc, example

Examples

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


## End(Not run)
```

`fastrml` *Rapidly converging algorithm for restricted maximum-likelihood (RML) estimation in linear mixed models*

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

Arguments

y	vector of responses. This is simply the individual yi 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(y1,y2,y3,y4) where length(y1)=2, length(y2)=3, length(y3)=2, and length(y4)=7. Then subj should be c(1,1,2,2,2,3,3,4,4,4,4,4,4,4,4).
pred	matrix of covariates used to predict y. The number of rows should be length(y). The first column will typically be constant (one), and the remaining columns correspond to other variables appearing in Xi and Zi.
xcol	vector of integers indicating which columns of pred will be used in Xi. That is, pred[,xcol] is the Xi matrices (stacked upon one another.
zcol	vector of integers indicating which columns of pred will be used in Zi. That is, pred[,zcol] is the Zi matrices (stacked upon one another).

`vmax` optional matrix of dimension c(max(occ),max(occ)) from which the Vi matrices will be extracted. In a longitudinal dataset, vmax would represent the Vi matrix for an individual with responses at all possible occasions 1,2,...,nmax=max(occ); for individuals with responses at only a subset of these occasions, the Vi 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 Vi = 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 nmax distinct occasions, each element of y corresponds to one subject-occasion, and the elements of occ 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".)

`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(length(zcol),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 all(abs(new-old)<eps*abs(old)).

Details

A full description of the algorithm is given in Section 3 of Schafer (1998). Scoring is carried out on log(sigma2) and the nonredundant elements of the inverse of psi/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

yi = Xi%*%beta + Zi%*%bi + ei , i=1,...,m,

where

yi = (ni x 1) response vector for subject or cluster i; Xi = (ni x p) matrix of covariates; Zi = (ni x q) matrix of covariates; beta = (p x 1) vector of coefficients common to the population (fixed effects); bi = (q x 1) vector of coefficients specific to subject or cluster i (random effects); and ei = (ni x 1) vector of residual errors.

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

bi $\sim$ N(0,psi) independently for i=1,...,m.

The residual vector ei is assumed to be

ei $\sim$ N(0,sigma2*Vi)

where Vi is a known (ni x ni) matrix. In most applications, Vi 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(length(zcol),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.
loglik	vector of length "iter" reporting the value of the loglikelihood at each iteration.
cov.beta	matrix of dimension c(length(xcol),length(xcol)) containing estimated variances and covariances for elements of "beta". These are conventional estimates which regard the variance parameters (sigma2 and psi) as fixed at their RML estimates.
b.hat	a matrix with length(zcol) rows and m columns, where b.hat[,i] is an empirical Bayes estimate of bi.
cov.b	an array of dimension length(zcol) by length(zcol) by m, where cov.b[,,i] is an empirical Bayes estimate of the covariance matrix associated with bi. These are conventional estimates which regard the variance parameters (sigma2 and psi) as fixed at their RML estimates.
cov.beta.new	matrix of dimension c(length(xcol),length(xcol)) containing estimated variances and covariances for elements of "beta". These are improved estimates which account for uncertainty in estimating the variance parameters (sigma2 and psi).
cov.b.new	an array of dimension length(zcol) by length(zcol) by m, where cov.b.new[,,i] is an estimated covariance matrix for bi. These are improved estimates which account for uncertainty in estimating the variance parameters (sigma2 and psi).
cov.b.beta.new	an array of dimension length(zcol) by length(xcol) by m, where cov.b.beta.new[,,i] contains the estimated covariances bewteen bi and beta. These are improved estimates which account for uncertainty in estimating the variance parameters (sigma2 and psi). Note that conventional estimates which regard sigma2 and psi as fixed assume zero covariance between each bi and beta.

References

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

See Also

ecmeml, ecmerml, fastml, fastmode, mgibbs, fastmcmc, example

Examples

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


## End(Not run)
```

`marijuana` *A pilot study of the clinical and psychological effects of maarijuana*

Description

Nine male subjects were given three treatments in the form of low-dose, high-dose, and placebo cigarettes. The order of treatments within subjects was balanced in a replicated 3 x 3 Latin square, but because the order for each subject was not reported in the article, (I shall proceed as if) the order effects are negligible. Changes in heart rate were recorded 15 and 90 minutes after marijuana use, and five of the 54 data values are missing.

Usage

```
data(marijuana)
```

Format

A data frame

Source

Wei AT, Zinberg NE, Nelson JM. Clinical and psychological effects of marijuana in man. Science 1968; 162:1234-1242

`mgibbs` *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" will perform better. This algorithm is provided mainly for comparison against "fastmcmc".

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

Usage

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

Arguments

y	vector of responses. This is simply the individual yi 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(y1,y2,y3,y4) where length(y1)=2, length(y2)=3, length(y3)=2, and length(y4)=7. Then subj should be c(1,1,2,2,2,3,3,4,4,4,4,4,4,4,4).
pred	matrix of covariates used to predict y. The number of rows should be length(y). The first column will typically be constant (one), and the remaining columns correspond to other variables appearing in Xi and Zi.
xcol	vector of integers indicating which columns of pred will be used in Xi. That is, pred[,xcol] is the Xi matrices (stacked upon one another.
zcol	vector of integers indicating which columns of pred will be used in Zi. That is, pred[,zcol] is the Zi matrices (stacked upon one another).
prior	A list with four components specifying the hyperparameters of the prior distribution applied to sigma2 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(length(zcol),length(zcol)).
seed	Seed for random number generator. This should be a positive integer.
vmax	optional matrix of dimension c(max(occ),max(occ)) from which the Vi matrices will be extracted. In a longitudinal dataset, vmax would represent the Vi matrix for an individual with responses at all possible occasions 1,2,...,nmax=max(occ); for individuals with responses at only a subset of these occasions, the Vi 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 Vi = 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 nmax distinct occasions, each element of y corresponds to one subject-occasion, and the elements of occ 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".)
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(length(zcol),length(zcol)), and "sigma2" should be a scalar.
iter	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

yi = Xi%*%beta + Zi%*%bi + ei , i=1,. . . ,m,

where

yi = (ni x 1) response vector for subject or cluster i; Xi = (ni x p) matrix of covariates; Zi = (ni x q) matrix of covariates; beta = (p x 1) vector of coefficients common to the population (fixed effects); bi = (q x 1) vector of coefficients specific to subject or cluster i (random effects); and ei = (ni x 1) vector of residual errors.

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

bi $\sim$ N(0,psi) independently for i=1,. . . ,m.

The residual vector ei is assumed to be

ei $\sim$ N(0,sigma2*Vi)

where Vi is a known (ni x ni) matrix. In most applications, Vi is the identity matrix.

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

sigma2 $\sim$ a / chisq(b),

where chisq(b) denotes a chisquare random variable with b degrees of freedom, and a and b are user-defined hyperparameters. Values for the hyperparmeters 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,

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

where psiinv 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 Dinv/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.

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

References

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

See Also

ecmeml, ecmerml, fastml, fastrml, fastmode, fastmcmc, example

Examples

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


## End(Not run)
```

Index