Package 'glmmEP'

September 8, 2026

Version 1.0-4

Date 2026-09-08

Title Generalized Linear Mixed Model Analysis via Expectation Propagation

Maintainer Matt P. Wand <matt.wand@uts.edu.au>

Depends stats

Imports lme4, matrixcalc

Suggests mlmRev

Description Approximate frequentist inference for generalized linear mixed model analysis with expectation propagation used to circumvent the need for multivariate integration. In this version, the random effects can be any reasonable dimension. However, only probit mixed models with one level of nesting are supported. The methodology is described in Hall, Johnstone, Ormerod, Wand and Yu (Journal of the American Statistical Association, 2020).

License GPL (>= 2)

NeedsCompilation yes

Author Matt P. Wand [aut, cre],
James C.F. Yu [aut],
Cleve Moler [ctb] (LINPACK routines in src)

Repository CRAN

Date/Publication 2026-09-08 02:50:02 UTC

Contents

glmmEP 2
glmmEP.control 4
glmmEPvignette 6
glmmSimData 6
summary.glmmEP 7

Index **9**

glmmEP *Approximate frequentist inference for generalized linear mixed models via expectation propagation.*

Description

The machine learning paradigm known as expectation propagation, typically used for approximation inference in Bayesian graphical models contexts, is adapted to the problem of frequentist inference in generalized linear mixed model analysis. The essence is rapid approximation evaluation of the log-likelihood surface by using expectation propagation projections onto the Multivariate Normal family that circumvent the need for numerical integration. The approach applies to any reasonable dimension of the random effects vector, but currently is limited to probit mixed models with one level of nesting. The reference below provides full details.

Usage

```
glmmEP(y,Xfixed=NULL,Xrandom,idNum,control)
```

Arguments

y	Vector containing the (0/1) response data.
Xfixed	Matrix with columns consisting of predictors that enter the model as fixed effects. Each entry of the first column equal the number 1 corresponding to the fixed effects intercept.
Xrandom	Matrix with columns consisting of predictors that enter the model as random effects. Each entry of the first column must equal the number 1 corresponding to the random effects intercept.
idNum	Vector containing group membership identification numbers.
control	Function for controlling convergence and other specifications.

Author(s)

Matt P. Wand <matt.wand@uts.edu.au> and James C.F. Yu <jamescfyu@gmail.com>

References

Hall, P., Johnstone, I.M., Ormerod, J.T., Wand, M.P. and Yu, J.C.F. (2020). Fast and accurate binary response mixed model analysis via expectation propagation. *Journal of the American Statistical Association*, **115**, 1902-1916.

Examples

```
library(glmmEP)

# Simulate data corresponding to
# a simple random intercept model:
```

```
set.seed(1)  ; m <- 100 ; n <- 10
beta0True <- 0.37 ; beta1True <- 0.93 ; sigmaTrue <- 0.73
uTrue <- rnorm(m)*sigmaTrue
x <- runif(m*n)
y <- rbinom(m*n,1,pnorm(beta0True + beta1True*x
            + crossprod(t(kronecker(diag(m),rep(1,n))),uTrue)))
idNum <- rep(1:m,each=n)
Xfixed <- cbind(1,x)
Xrandom <- matrix(1,length(y),1)
colnames(Xfixed) <- c("intercept","x")

# Obtain and summarise glmmEP() fit:

fit <- glmmEP(y,Xfixed,Xrandom,idNum)
summary(fit)

# Obtain simulated data corresponding to the
# simulation study in Section 4.1.2 of
# Hall et al. (2020):

dataObj <- glmmSimData(seed=54321)
y <- dataObj$y
Xfixed <- dataObj$Xfixed
Xrandom <- dataObj$Xrandom
idNum <- dataObj$idNum

# Obtain the probit mixed model fit and summaries:


fitSimData <- glmmEP(y,Xfixed,Xrandom,idNum)
summary(fitSimData)
plot(fitSimData$randomEffects) ; abline(h=0) ; abline(v=0)


if (require("mlmRev"))
{
   # Example involving the data frame `Contraception'
   # in the package `mlmRev', starting with
   # setting up the input vectors and matrices
   # for a random intercepts model:

   data(Contraception)
   y <- as.numeric(as.character(Contraception$use)=="Y")
   Xfixed <- cbind(1,as.numeric(as.character(Contraception$urban)=="Y"),
                   Contraception$age,
                   as.numeric(as.character(Contraception$livch)=="1"),
                   as.numeric(as.character(Contraception$livch)=="2"),
                   as.numeric(as.character(Contraception$livch)=="3+"))
   colnames(Xfixed) <- c("intercept","isUrban","age","livChEq1",
                         "livChEq2","livChGe3")
   Xrandom <- as.matrix(rep(1,length(y)))
   colnames(Xrandom) <- "intercept"
   idNum <- as.numeric(as.character(Contraception$district))
```

```
# Obtain random intercepts probit mixed model fit and summaries:


fitContracRanInt <- glmmEP(y,Xfixed,Xrandom,idNum)
summary(fitContracRanInt)
hist(as.numeric(fitContracRanInt$randomEffects),
     xlab="random intercepts predicted values",probability=TRUE,
     col="dodgerblue",breaks=15,main="")
abline(v=0,col="slateblue",lwd=2)


# Change `Xrandom' to correspond to a random intercepts and
# slopes model:

Xrandom <- cbind(1,as.numeric(as.character(Contraception$urban)=="Y"))
colnames(Xrandom) <- c("intercept","isUrban")

# Obtain random intercepts and slopes probit mixed model fit and summaries:


fitContracRanIntAndSlp <- glmmEP(y,Xfixed,Xrandom,idNum)
summary(fitContracRanIntAndSlp)
plot(fitContracRanIntAndSlp$randomEffects,
     col="dodgerblue",xlab="random intercepts predicted values",
     ylab="random slopes predicted values",bty="l")
abline(v=0,col="slateblue",lwd=2); abline(h=0,col="slateblue",lwd=2)

}
```

`glmmEP.control` *Controlling generalized linear mixed model fitting via expectation propagation*

Description

Function for optional use in calls to `glmmEP()` to control convergence values and other specifications for expectation propagation-based fitting of generalized linear mixed models.

Usage

```
glmmEP.control(confLev=0.95,BFGSmaxit=500,BFGSreltol=1e-10,
               EPmaxit=100,EPreltol=1e-5,NMmaxit=100,NMreltol=1e-10,
               quiet=FALSE,preTransfData=TRUE)
```

Arguments

`confLev`	Confidence level of confidence intervals expressed as a proportion (i.e. a number between 0 and 1). The default is 0.95 corresponding to 95% confidence intervals.

BFGSmaxit	Positive integer specifying the maximum number of iterations in the Broyden-Fletcher-Goldfarb-Shanno optimization phase. The default is 500.
BFGSreltol	Positive number specifying the relative tolerance value as defined in the R function optim() in the Broyden-Fletcher-Goldfarb-Shanno optimization phase. The default is 1e-10.
EPmaxit	Positive integer specifying the maximum number of iterations in the expectation propagation message passing iterations. The default is 100.
EPreltol	Positive number specifying the relative tolerance value for the expectation propagation message passing iterations. The default is 1e-5.
NMmaxit	Positive integer specifying the maximum number of iterations in the Nelder-Mead optimization phase. The default is 100.
NMreltol	Positive number specifying the relative tolerance value as defined in the R function optim() in the Nelder-Mead optimization phase. The default is 1e-10.
quiet	Flag for specifying whether or not glmmEP() runs 'quietly' or with progress reports printed to the screen. The default is FALSE.
preTransfData	Flag for specifying whether or not the predictor data are pre-transformed to the unit interval for fitting, with estimates, predictions and confidence intervals transformed to match the units of the original data before. The default is TRUE.

Value

A list containing values of each of the fifteen control parameters, packaged to supply the `control` argument to `glmmEP`. The values for `glmmEP.control` can be specified in the call to `glmmEP`.

Author(s)

Matt P. Wand <matt.wand@uts.edu.au> and James C.F. Yu <jamescfyu@gmail.com>

References

Hall, P., Johnstone, I.M., Ormerod, J.T., Wand, M.P. and Yu, J.C.F. (2020). Fast and accurate binary response mixed model analysis via expectation propagation. *Journal of the American Statistical Association*, **115**, 1902-1916.

Examples

```
library(glmmEP)

# Obtain simulated data corresponding to the simulation study in Section 4.1.2. of
# Hall et al. (2020):

dataObj <- glmmSimData(seed=54321)
y <- dataObj$y
Xfixed <- dataObj$Xfixed
Xrandom <- dataObj$Xrandom
idNum <- dataObj$idNum

# Obtain and summarise probit mixed model fit with user control
```

```
# of some of the parameters in glmmEP.control():

myNMmaxit <- 500 ; myBFGSreltol <- 0.001

fitSimData <- glmmEP(y,Xfixed,Xrandom,idNum,
           control=glmmEP.control(NMmaxit=myNMmaxit,BFGSreltol=myBFGSreltol,quiet=TRUE))
summary(fitSimData)
```

---

glmmEPvignette *Display the package's vignette.*

Description

The vignette of the glmmEP package is displayed using the default PDF file browser. It provides a detailed detailed description of use of the package for binary mixed model analysis.

Usage

```
glmmEPvignette()
```

Author(s)

Matt P. Wand <matt.wand@uts.edu.au> and James C.F. Yu <jamescfyu@gmail.com>

Examples

```
glmmEPvignette()
```

---

glmmSimData *Simulation of data from a generalized linear mixed model.*

Description

Section 4.1.2 of the refence below descries a simulation study with data generated from a probit mixed model with six fixed effects parameters and a bivariate random effects vector having a 2 by 2 symmetric positive definite covariance matrix. The function simulates a data set from this model with 2500 groups and the number of observation in each group being a random draw from {20,21,...,30}.

Usage

```
glmmSimData(seed = 12345)
```

Arguments

seed	A positive integer which acts the seed for random data generation.

Author(s)

Matt P. Wand <matt.wand@uts.edu.au> and James C.F. Yu <jamescfyu@gmail.com>

References

Hall, P.,Johnstone, I.M., Ormerod, J.T., Wand, M.P. and Yu, J.C.F. (2020). Fast and accurate binary response mixed model analysis via expectation propagation. *Journal of the American Statistical Association*, **115**, 1902-1916.

Examples

```
# Obtain simulated data corresponding to the simulation study in Section 4.1.2. of
# Hall et al. (2020):

library(glmmEP)
dataObj <- glmmSimData(seed = 12345)
print(names(dataObj))
```

`summary.glmmEP` *Inferential summary for a generalized linear mixed model with expectation propagation fitting.*

Description

Produces a table containing approximated maximum likelihood estimates and corresponding confidence interval limits for the fixed effect parameters and random effects standard deviation and correlation parameters.

Usage

```
## S3 method for class 'glmmEP'
summary(object,...)
```

Arguments

`object`	A `glmmEP()` fit object.
`...`	Place-holder for additional arguments.

Author(s)

Matt P. Wand <matt.wand@uts.edu.au> and James C.F. Yu <jamescfyu@gmail.com>

Examples

```
dataObj <- glmmSimData(seed=54321)
y <- dataObj$y
Xfixed <- dataObj$Xfixed
Xrandom <- dataObj$Xrandom
idNum <- dataObj$idNum

fitSimData <- glmmEP(y,Xfixed,Xrandom,idNum)
summary(fitSimData)
```

Index

glmmEP, 2
glmmEP.control, 4
glmmEPvignette, 6
glmmSimData, 6

summary.glmmEP, 7