

Package ‘MMS’

May 17, 2018

Type Package

Title Fixed Effects Selection in Linear Mixed Models

Version 3.0.11

Date 2018-05-17

Author F.Rohart

Maintainer Florian Rohart <florian.rohart@gmail.com>

Description Perform Fixed effects Selection in Linear Mixed Models using a multicycle ECM algorithm; see F. Rohart & al, 2014, <doi:10.1016/j.csda.2014.06.022>.

License GPL-3

Depends glmnet,Matrix, mht (>= 3.00)

NeedsCompilation no

Repository CRAN

Date/Publication 2018-05-17 07:33:55 UTC

R topics documented:

| | |
|-----------------------|-----------|
| MMS-package | 1 |
| lassop | 3 |
| lmme | 5 |
| mhtp | 7 |
| refit.mhtp | 9 |
| Index | 13 |

MMS-package *Fixed Effects Selection in Linear Mixed Models*

Description

Perform Fixed effects Selection in Linear Mixed Models using a multicycle ECM algorithm; see F. Rohart & al, 2014, <doi:10.1016/j.csda.2014.06.022>.

Details

The DESCRIPTION file:

```

Package:      MMS
Type:         Package
Title:        Fixed Effects Selection in Linear Mixed Models
Version:      3.0.11
Date:         2018-05-17
Author:       F.Rohart
Maintainer:   Florian Rohart <florian.rohart@gmail.com>
Description:  Perform Fixed effects Selection in Linear Mixed Models using a multicycle ECM algorithm; see F. Rohart & al
License:      GPL-3
Depends:      glmnet,Matrix, mht (>= 3.00)
Packaged:    2018-05-17 03:56:36 UTC; florian

```

Index of help topics:

| | |
|-------------|--|
| MMS-package | Fixed Effects Selection in Linear Mixed Models |
| lassop | L1-penalization in linear mixed models |
| lmme | Linear mixed model estimation |
| mhtp | Multiple testing procedure for variable selection in linear mixed models |
| refit.mhtp | Multiple testing procedure for variable selection in linear mixed models |

Two major functions:

[lassop](#) performs fixed effects selection in linear mixed models through a L1-penalization of the log-likelihood of the marginal model. The method optimizes a criterion via a multicycle ECM algorithm.

[mhtp](#) performs fixed effects selection in linear mixed models through a modification of the previous algorithm which allows the use of any variable selection method developed in linear models. It is a combination of the [mht](#) method from the [mht-package](#) and the modified algorithm.

Author(s)

F.Rohart

Maintainer: Florian Rohart <florian.rohart@gmail.com>

References

Multiple hypotheses testing for variable selection; F. Rohart, 2016, <https://doi.org/10.1111/anzs.12157>
 Selection of fixed effects in high dimensional linear mixed models using a multicycle ECM algorithm; F. Rohart & al, 2014, <https://doi.org/10.1016/j.csga.2014.06.022>

Examples

```

## Not run:
N <- 20          # number of groups
p <- 20          # number of covariates (including intercept)
q <- 2           # number of random effect covariates
ni <- rep(6,N)   # observations per group
n <- sum(ni)     # total number of observations

grp <- factor(rep(1:N,ni)) # grouping variable
grp=rbind(grp,grp)

beta <- c(1,2,4,3,rep(0,p-3)) # fixed-effects coefficients
x <- cbind(1,matrix(rnorm(n*p),nrow=n)) # design matrix

u1=rnorm(N,0,sd=sqrt(2))
u2=rnorm(N,0,sd=sqrt(2))
bi1 <- rep(u1,ni)
bi2 <- rep(u2,ni)
bi <- rbind(bi1,bi2)

z=x[,1:2,drop=FALSE]

epsilon=rnorm(120)
y <- numeric(n)
for (k in 1:n) y[k] <- x[k,]%*%beta + t(z[k,])%*%bi[,k] + epsilon[k]
#####
fit0=lme(x[,1:4],y,z,grp)
fit1=lassop(x,y,z,grp,mu=0.2,fix=1,rand=c(1,2))
#fit1=lassop(x,y,z,grp,mu=0.2,fix=1,rand=c(1,2),showit=TRUE)

fit2=mhpt(x,y,z,grp,fix=1,rand=c(1,2),alpha=0.1,num=15)
#fit2=mhpt(x,y,z,grp,fix=1,rand=c(1,2),alpha=0.1,num=15,show=c(1,1,1))

## End(Not run)

```

lassop

*L1-penalization in linear mixed models***Description**

Performs a L1-penalization in linear mixed models

Usage

```
lassop(data,Y,z,grp,D,mu,step,fix,rand,penalty.factor,alpha,showit)
```

Arguments

| | |
|-----------------------------|--|
| <code>data</code> | Input matrix of dimension $n * p$; each row is an observation vector. The intercept should be included in the first column as $(1, \dots, 1)$. If not, it is added. |
| <code>Y</code> | Response variable of length n . |
| <code>z</code> | Random effects matrix. Of size $n * q$. |
| <code>grp</code> | Grouping variable of length n . |
| <code>D</code> | Logical value. If TRUE, the random effects are considered to be independent, i.e. Ψ is a diagonal matrix. <code>D=TRUE</code> should be used with nested grouping factors. |
| <code>mu</code> | Positive regularization number to be used for the Lasso. |
| <code>step</code> | The algorithm performs at most <code>step</code> iterations. Default is 3000. |
| <code>fix</code> | Number of variables which are not submitted to selection. They have to be in the first columns of <code>data</code> . Default is 1, the selection is not performed on the intercept. |
| <code>rand</code> | A vector of length q : each entry k is the position of the random effects number k in the data matrix, 0 otherwise. If <code>z</code> contains variables that have both a fixed and a random effect, it is advised to not submit them to selection. |
| <code>penalty.factor</code> | Argument of 'glmnet'. Separate penalty factors can be applied to each coefficient. This is a number that multiplies λ to allow differential shrinkage. Can be 0 for some variables, which implies no shrinkage, and that variable is always included in the model. Default is 1 for all variables that are not in <code>1:fix</code> . |
| <code>alpha</code> | Argument of 'glmnet'. The elasticnet mixing parameter, with $0 \leq \alpha \leq 1$. <code>alpha=1</code> is the lasso penalty, and <code>alpha=0</code> the ridge penalty. |
| <code>showit</code> | Logical value. If TRUE, shows the iterations of the algorithm. Default is FALSE. |

Details

This function performs fixed effects selection in linear mixed models through a L1-penalization of the log-likelihood of the marginal model. The method optimizes a criterion via a multicycle ECM algorithm at the regularization parameter μ .

Two algorithms are available: one when the random effects are assumed to be independent (`D=TRUE`) and one when they are not (`D=FALSE`). Selection on the random is only performed when `D=TRUE`.

Value

A 'lassop' object is returned.

| | |
|----------------------------|--|
| <code>data</code> | List of the user-data: the scaled matrix used in the algorithm, the first column being $(1, \dots, 1)$; <code>Y</code> and <code>Z</code> , which is the design matrix of the random effects. |
| <code>beta</code> | Estimation of the fixed effects. |
| <code>fitted.values</code> | Fitted values calculated with the fixed effects and the random effects. |
| <code>Psi</code> | Variance of the random effects. Matrix of dimension $q * q$. |
| <code>sigma_e</code> | Variance of the residuals. |

| | |
|----------|---|
| it | Number of iterations of the algorithm. |
| converge | Logical. TRUE if the algorithm has converged, FALSE otherwise. |
| u | Vector of the concatenation of the estimated random effects (u_1',...,u_q')'. |
| call | The call that produced this object. |
| mu | The penalty used in the algorithm. |

Examples

```
## Not run:
N <- 20          # number of groups
p <- 80          # number of covariates (including intercept)
q <- 2           # number of random effect covariates
ni <- rep(6,N)   # observations per group
n <- sum(ni)     # total number of observations

grp <- factor(rep(1:N,ni)) # grouping variable
grp=rbind(grp,grp)

beta <- c(1,2,4,3,rep(0,p-3)) # fixed-effects coefficients
x <- cbind(1,matrix(rnorm(n*p),nrow=n)) # design matrix

u1=rnorm(N,0,sd=sqrt(2))
u2=rnorm(N,0,sd=sqrt(2))
bi1 <- rep(u1,ni)
bi2 <- rep(u2,ni)
bi <- rbind(bi1,bi2)

z=x[,1:2,drop=FALSE]

epsilon=rnorm(120)
y <- numeric(n)
for (k in 1:n) y[k] <- x[k,]%*%beta + t(z[k,])%*%bi[,k] + epsilon[k]

#####

#independent random effects
fit=lassop(x,y,z,grp,D=1,mu=0.2,fix=1,rand=c(1,2))

#dependent random effects
fit=lassop(x,y,z,grp,mu=0.2,fix=1,rand=c(1,2))

## End(Not run)
```

Description

Estimation of fixed and random effects in linear mixed models

Usage

```
lmme(data, Y, z, grp, D, step, showit)
```

Arguments

| | |
|--------|---|
| data | Input matrix of dimension $n * p$; each row is an observation vector. The intercept should be included in the first column as (1,...,1). If not, it is added. |
| Y | Response variable of length n . |
| z | Random effects matrix. Of size $n*q$. |
| grp | Grouping variable of length n . |
| D | Logical value. If TRUE, the random effects are considered to be independent, i.e. Psi is a diagonal matrix. D=TRUE should be used with nested grouping factors. |
| step | The algorithm performs at most step iterations. Default is 3000. |
| showit | Logical value. If TRUE, shows the convergence process of the algorithm. Default is FALSE. |

Details

lmme performs an ML-estimation of fixed and random effects in linear mixed models when no selection is involved. Two algorithms are available: one when the random effects are assumed to be independent (D=TRUE) and one when they are not (D=FALSE).

Value

| | |
|---------------|---|
| data | List of the user-data: the scaled matrix used in the algorithm, the first column being (1,...,1); Y; z and grp. |
| beta | Estimation of the selected fixed effects. |
| Psi | Variance of the random effects. Matrix of dimension $q*q$. |
| sigma_e | Variance of the noise. |
| fitted.values | Fitted values calculated with the fixed effects and the random effects. |
| it | Number of iterations of the algorithm. |
| converge | Did the algorithm converge? |
| u | Vector of the concatenation of the estimated random effects (u_1', \dots, u_q'). |
| call | The call that produced this object. |

Examples

```
## Not run:
N <- 20          # number of groups
p <- 80          # number of covariates (including intercept)
q <- 2           # number of random effect covariates
ni <- rep(6,N)  # observations per group
n <- sum(ni)    # total number of observations
```

```

grp <- factor(rep(1:N,ni)) # grouping variable
grp=rbind(grp,grp)

beta <- c(1,2,4,3,rep(0,p-3)) # fixed-effects coefficients
x <- cbind(1,matrix(rnorm(n*p),nrow=n)) # design matrix

u1=rnorm(N,0,sd=sqrt(2))
u2=rnorm(N,0,sd=sqrt(2))
bi1 <- rep(u1,ni)
bi2 <- rep(u2,ni)
bi <- rbind(bi1,bi2)

z=x[,1:2,drop=FALSE]

epsilon=rnorm(120)
y <- numeric(n)
for (k in 1:n) y[k] <- x[k,]%*%beta + t(z[k,])%*%bi[,k] + epsilon[k]

#####
fit=lmme(x,y,z,grp)

## End(Not run)

```

mhtp

Multiple testing procedure for variable selection in linear mixed models

Description

Performs a multiple hypotheses testing in linear mixed models

Usage

```
mhtp(data,Y,z,grp,D,fix,rand,alpha,step,num,ordre,m,show,IT,maxq,speed)
```

Arguments

| | |
|------|--|
| data | Input matrix of dimension $n * p$; each row is an observation vector. The intercept should be included in the first column as $(1, \dots, 1)$. If not, it is added. |
| Y | Response variable of length n . |
| z | Random effects matrix. Of size $n * q$. |
| grp | Grouping variable of length n . |
| D | Logical value. If TRUE, the random effects are considered to be independent, i.e. Ψ is a diagonal matrix. $D=TRUE$ should be used with nested grouping factors. |
| fix | Number of variables which are not submitted to selection. They have to be in the first columns of data. Default is 1, the selection is not performed on the intercept. |

| | |
|-------|--|
| rand | A vector of length q: each entry k is the position of the random effects number k in the data matrix, 0 otherwise. If z contains variables that have both a fixed and a random effect, it is advised to not submit them to selection. |
| alpha | A user supplied type I error sequence. Default is (0.1,0.05). |
| step | The algorithm performs at most step iterations. Default is 3000. |
| num | Number of variables one wishes to order. Default is min(n-1,p-1,30). |
| ordre | Several possible algorithms to order the variables, ordre=c("bolasso", "pval", "pval_hd", "FR"). "bolasso" uses the dyadic algorithm with the Bolasso technique, "pval" uses the p-values obtained with a regression on the full set of variables (only when p<n), "pval_hd" uses marginal regression, "FR" uses Forward Regression. Default is "bolasso". |
| m | Number of bootstrapped iteration of the Lasso. Only use if the algorithm is set to "bolasso". Default is m=10. |
| show | Vector of logical values, show=(showordre, showresult, showit). Default is (0,0,0). If showordre=TRUE, show the ordered variables at each step of the algorithm. if showresult=TRUE, show the value of the statistics and the estimated quantile at each step of the procedure; if ordre=bolasso. if showit=TRUE, show the iterations of the algorithm. |
| IT | Number of simulations in the calculation of the quantile. Default is 10000. |
| maxq | Number of maximum multiple hypotheses testing to do. Default is min(log(min(n,p)-1,2),5). |
| speed | Logical value. If TRUE, the algorithm is speeded up once the criteria convergence in beta and u are fulfilled. |

Details

mhtp performs fixed effects selection in linear mixed models. It is a combination of the [mht](#) function from the [mht-package](#) with the algorithm used in the [lassop](#) function; so you might want to have a look at the help of [mht](#).

Value

A 'mhtp' object is returned for which `refit` is available.

| | |
|---------------|---|
| data | List of the user-data: the scaled matrix used in the algorithm, the first column being (1,...,1); Y; z and grp. |
| beta | The estimated vector of fixed effects coefficients. Each row concern a specific user level alpha. |
| fitted.values | Fitted values calculated with the fixed effects and the random effects. |
| u | Matrix with #alpha columns. Each column is the concatenation of the estimated random effects (u_1', ..., u_q') |
| Psi | Variance of the random effects. Matrix of dimension q*q. |
| sigma_e | Variance of the residuals. |
| it | Number of iterations of the algorithm. |

| | |
|-----------|---|
| quantile | Array of all the estimated quantiles calculated during the procedure. |
| ordrebeta | All different order that has been used during the procedure. |
| converge | Did the algorithm converge? |
| call | The call that produced this object. |
| arg | List of all the arguments of the function (used to refit the function). |

See Also

[mht](#), [refit.mhtp](#)

Examples

```
## Not run:
N <- 20          # number of groups
p <- 20         # number of covariates (including intercept)
q <- 2          # number of random effect covariates
ni <- rep(6,N)  # observations per group
n <- sum(ni)    # total number of observations

grp <- factor(rep(1:N,ni)) # grouping variable
grp=rbind(grp,grp)

beta <- c(1,2,4,3,rep(0,p-3)) # fixed-effects coefficients
x <- cbind(1,matrix(rnorm(n*p),nrow=n)) # design matrix

u1=rnorm(N,0,sd=sqrt(2))
u2=rnorm(N,0,sd=sqrt(2))
bi1 <- rep(u1,ni)
bi2 <- rep(u2,ni)
bi <- rbind(bi1,bi2)

z=x[,1:2,drop=FALSE]

epsilon=rnorm(120)
y <- numeric(n)
for (k in 1:n) y[k] <- x[k,]%*%beta + t(z[k,])%*%bi[,k] + epsilon[k]

#####
fit=mhtp(x,y,z,grp,D=0,fix=1,rand=c(1,2),alpha=0.1,num=15)
#fit=mhtp(x,y,z,grp,D=0,fix=1,rand=c(1,2),alpha=0.1,num=15,show=c(1,1,1))

## End(Not run)
```

Description

Performs a multiple hypotheses testing in linear mixed models

Usage

```
## S3 method for class 'mhpt'
refit(object, Ynew, z, grp, D, fix, rand, alpha, step, num, ordre, m, show, IT, maxq, speed, ...)
```

Arguments

| | |
|--------|--|
| object | Object of class "mhpt". |
| Ynew | Response variable of length n. |
| z | Random effects matrix. Of size n*q. |
| grp | Grouping variable of length n. |
| D | Logical value. If TRUE, the random effects are considered to be independent, i.e. Psi is a diagonal matrix. D=TRUE should be used with nested grouping factors. |
| fix | Number of variables which are not submitted to selection. They have to be in the first columns of data. Default is 1, the selection is not performed on the intercept. |
| rand | A vector of length q: each entry k is the position of the random effects number k in the data matrix, 0 otherwise. If z contains variables that have both a fixed and a random effect, it is advised to not submit them to selection. |
| alpha | A user supplied type I error sequence. Default is (0.1,0.05). |
| step | The algorithm performs at most step iterations. Default is 3000. |
| num | Number of variables one wishes to order. Default is min(n-1,p-1,30). |
| ordre | Several possible algorithms to order the variables, ordre=c("bolasso", "pval", "pval_hd", "FR"). "bolasso" uses the dyadic algorithm with the Bolasso techniaue, "pval" uses the p-values obtained with a regression on the full set of variables (only when p<n), "pval_hd" uses marginal regression, "FR" uses Forward Regression. Default is "bolasso". |
| m | Number of bootstrapped iteration of the Lasso. Only use if the algorithm is set to "bolasso". Default is m=10. |
| show | Vector of logical values, show=(showordre,showresult,showit). Default is (0,0,0). If showordre==TRUE, show the ordered variables at each step of the algorithm. if showresult==TRUE, show the value of the statistics and the estimated quantile at each step of the procedure; if ordre=bolasso. if showit==TRUE, show the iterations of the algorithm. |
| IT | Number of simulations in the calculation of the quantile. Default is 10000. |
| maxq | Number of maximum multiple hypotheses testing to do. Default is min(log(min(n,p)-1,2),5). |
| speed | Logical value. If TRUE, the algorithm is speeded up once the criterion convergence in beta and u is fulfilled. |
| ... | not used |

Details

See [mhnp](#) for more details.

Value

A 'mhnp object' is returned.

| | |
|---------------|---|
| data | List of the user-data: the scaled matrix used in the algorithm, the first column being (1,...,1); Y; z and grp. |
| beta | Estimation of the selected fixed effects. |
| fitted.values | Fitted values calculated with the fixed effects and the random effects. |
| u | Matrix with #alpha columns. Each column is the concatenation of the estimated random effects (u_1',...,u_q') |
| Psi | Variance of the random effects. Matrix of dimension q*q. |
| sigma_e | Variance of the noise. |
| it | Number of iterations of the algorithm. |
| quantile | Array of all the estimated quantiles calculated during the procedure. |
| ordrebeta | All different order that has been used during the procedure. |
| converge | Did the algorithm converge? |
| call | The call that produced this object. |
| arg | List of all the arguments of the function. |

See Also

[mhnp](#)

Examples

```
## Not run:
N <- 20          # number of groups
p <- 20         # number of covariates (including intercept)
q <- 2          # number of random effect covariates
ni <- rep(6,N)  # observations per group
n <- sum(ni)    # total number of observations

grp <- factor(rep(1:N,ni)) # grouping variable
grp=rbind(grp,grp)

beta <- c(1,2,4,3,rep(0,p-3)) # fixed-effects coefficients
x <- cbind(1,matrix(rnorm(n*p),nrow=n)) # design matrix

u1=rnorm(N,0,sd=sqrt(2))
u2=rnorm(N,0,sd=sqrt(2))
bi1 <- rep(u1,ni)
bi2 <- rep(u2,ni)
bi <- rbind(bi1,bi2)
```

```
z=x[,1:2,drop=FALSE]

epsilon=rnorm(120)
y <- numeric(n)
for (k in 1:n) y[k] <- x[k,]%*%beta + t(z[k,])%*%bi[,k] + epsilon[k]

#####
fit=mhnp(x,y,z,grp,D=0,fix=1,rand=c(1,2),alpha=0.1,num=15)
fit2=refit(fit,Ynew=y)

## End(Not run)
```

Index

lassop, [2](#), [3](#), [8](#)
lmme, [5](#)

mht, [2](#), [8](#), [9](#)
mht-package, [8](#)
mhtp, [2](#), [7](#), [11](#)
MMS (MMS-package), [1](#)
MMS-package, [1](#)

refit.mhtp, [9](#), [9](#)