Package ‘EMMREML’

July 22, 2015

Type Package

Version 3.1

Date 2015-07-20

Title Fitting Mixed Models with Known Covariance Structures

Author Deniz Akdemir, Okeke Uche Godfrey

Maintainer Deniz Akdemir <deniz.akdemir.work@gmail.com>

Depends Matrix, stats

Description The main functions are 'emmreml', and 'emmremlMultiKernel'. 'emmreml' solves a mixed model with known covariance structure using the 'EMMA' algorithm. 'emmremlMultiKernel' is a wrapper for 'emmreml' to handle multiple random components with known covariance structures. The function 'emmremlMultivariate' solves a multivariate gaussian mixed model with known covariance structure using the 'ECM' algorithm.

License GPL-2

NeedsCompilation no

Repository CRAN

Date/Publication 2015-07-22 05:52:07

R topics documented:

EMMREML ................................................................. 2
emmreml ................................................................. 2
emmremlMultiKernel .................................................. 4
emmremlMultivariate .................................................... 6

Index 9
### Description

The main functions are emmreml, and emmremlMultiKernel. emmreml solves a mixed model with known covariance structure using the EMMA algorithm in Kang et al. (2008). emmremlMultiKernel is a wrapper for emmreml to handle multiple random components with known covariance structures. The function emmremlMultivariate solves a multivariate gaussian mixed model with known covariance structure using the ECM algorithm in Zhou and Stephens (2012).

### Details

Package: EMMREML  
Type: Package  
Version: 3.1  
Date: 2015-07-20  
License: GPL-2

### Author(s)

Deniz Akdemir, Okeke Uche Godfrey  
Maintainer: Deniz Akdemir <deniz.akdemir.work@gmail.com>

### References


### emmreml

**Solver for Gaussian mixed model with known covariance structure.**

### Description

This function estimates the parameters of the model

$$y = X\beta + Zu + e$$
where \( y \) is the \( n \) vector of response variable, \( X \) is a \( n \times q \) known design matrix of fixed effects, \( Z \) is a \( n \times l \) known design matrix of random effects, \( \beta \) is \( q \times 1 \) vector of fixed effects coefficients and \( u \) and \( e \) are independent variables with \( N_l(0, \sigma_u^2 K) \) and \( N_n(0, \sigma_e^2 I_n) \) correspondingly. It also produces the BLUPs and some other useful statistics like large sample estimates of variances and PEV.

Usage

```
emnreml(y, X, Z, K, varbetahat=FALSE, varuhat=FALSE, PEVhat=FALSE, test=FALSE)
```

Arguments

- \( y \) \( n \times 1 \) numeric vector
- \( X \) \( n \times q \) matrix
- \( Z \) \( n \times l \) matrix
- \( K \) \( l \times l \) matrix of known relationships
- varbetahat TRUE or FALSE
- varuhat TRUE or FALSE
- PEVhat TRUE or FALSE
- test TRUE or FALSE

Value

- \( Vu \) Estimate of \( \sigma_u^2 \)
- \( Ve \) Estimate of \( \sigma_e^2 \)
- betahat BLUEs for \( \beta \)
- uhat BLUPs for \( u \)
- \( \chi^2 \) test statistics for testing whether the fixed effect coefficients are equal to zero.
- pvalbeta pvalues obtained from large sample theory for the fixed effects. We report the pvalues adjusted by the "padjust" function for all fixed effect coefficients.
- \( \chi^2 \) test statistic values for testing whether the BLUPs are equal to zero.
- pvalu pvalues obtained from large sample theory for the BLUPs. We report the pvalues adjusted by the "padjust" function.
- varuhat Large sample variance for the BLUPs.
- varbetahat Large sample variance for the \( \beta \)'s.
- PEVhat Prediction error variance estimates for the BLUPs.
- loglik loglikelihood for the model.

Examples

```
n=200
M1<-matrix(rnorm(n*300), nrow=n)
K1<-cov(t(M1))
K1=K1/mean(diag(K1))
```
emmmremlMultiKernel

Function to fit Gaussian mixed model with multiple mixed effects with known covariances.

Description

This function is a wrapper for the emmreml to fit Gaussian mixed model with multiple mixed effects with known covariances. The model fitted is $y = X\beta + Z_1u_1 + Z_2u_2 + ...Z_ku_k + e$ where $y$ is the $n$ vector of response variable, $X$ is a $nxq$ known design matrix of fixed effects, $Z_j$ is a $nxl_j$ known design matrix of random effects for $j = 1, 2, ..., k$, $\beta$ is $nx1$ vector of fixed effects coefficients and $U = (u_1^t, u_2^t, ..., u_k^t)^t$ and $e$ are independent variables with $N_L(0, \text{blockdiag}(\sigma^2_{u_1}K_1, \sigma^2_{u_2}K_2, ..., \sigma^2_{u_k}K_k))$ and $N_n(0, \sigma^2_eI_n)$ correspondingly. The function produces the BLUPs for the $L = l_1 + l_2 + ... + l_k$ dimensional random effect $U$. The variance parameters for random effects are estimated as $(\hat{\omega}_1, \hat{\omega}_2, ..., \hat{\omega}_k) \ast \hat{\sigma}^2_e$ where $w = (w_1, w_2, ..., w_k)$ are the kernel weights. The function also provides some useful statistics like large sample estimates of variances and PEV.

Usage

emmmremlMultiKernel(y, X, Zlist, Klist, varbetahat=FALSE, varuhat=FALSE, PEVhat=FALSE, test=FALSE)

Arguments

<table>
<thead>
<tr>
<th>y</th>
<th>$nx1$ numeric vector</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>$nxq$ matrix</td>
</tr>
<tr>
<td>Zlist</td>
<td>list of random effects design matrices of dimensions $nxl_1, ..., nxl_k$</td>
</tr>
<tr>
<td>Klist</td>
<td>list of known relationship matrices of dimensions $l_1xl_1, ..., l_kxl_k$</td>
</tr>
<tr>
<td>varbetahat</td>
<td>TRUE or FALSE</td>
</tr>
<tr>
<td>varuhat</td>
<td>TRUE or FALSE</td>
</tr>
<tr>
<td>PEVhat</td>
<td>TRUE or FALSE</td>
</tr>
<tr>
<td>test</td>
<td>TRUE or FALSE</td>
</tr>
</tbody>
</table>
Value

- $Vu$: Estimate of $\sigma_u^2$
- $Ve$: Estimate of $\sigma_e^2$
- $\text{betahat}$: BLUEs for $\beta$
- $\text{uhat}$: BLUPs for $u$
- weights: Estimates of kernel weights
- $Xsq\text{testbeta}$: A $\chi^2$ test statistic based for testing whether the fixed effect coefficients are equal to zero.
- $pval\text{beta}$: p-values obtained from large sample theory for the fixed effects. We report the p-values adjusted by the "padjust" function for all fixed effect coefficients.
- $Xsq\text{testu}$: A $\chi^2$ test statistic based for testing whether the BLUPs are equal to zero.
- $pvalu$: p-values obtained from large sample theory for the BLUPs. We report the p-values adjusted by the "padjust" function.
- var$\text{uhat}$: Large sample variance for the BLUPs.
- var$\text{betahat}$: Large sample variance for the $\beta$'s.
- PE$\text{Vuhat}$: Prediction error variance estimates for the BLUPs.
- loglik: Loglikelihood for the model.

Examples

```r
### example
# Data from Gaussian process with three
# (total four, including residuals) independent
# sources of variation

n=80
M1<-matrix(rnorm(n*10), nrow=n)
M2<-matrix(rnorm(n*20), nrow=n)
M3<-matrix(rnorm(n*5), nrow=n)

# Relationship matrices
K1<-cov(t(M1))
K2<-cov(t(M2))
K3<-cov(t(M3))
K1=K1/mean(diag(K1))
K2=K2/mean(diag(K2))
K3=K3/mean(diag(K3))

# Generate data
covY<-2*(.2*K1+.7*K2+.1*K3)+diag(n)
Y<-10+crossprod(chol(covY), rnorm(n))
```
#training set
Trainsamp<-sample(1:80, 60)

funout<-emmremlMultiKernel(y=Y[Trainsamp], X=matrix(rep(1, n)[Trainsamp], ncol=1),
Zlist=list(diag(n)[Trainsamp,], diag(n)[Trainsamp,], diag(n)[Trainsamp,]),
Klist=list(K1, K2, K3),
varbetahat=FALSE, varuhat=FALSE, PEVwhat=FALSE, test=FALSE)
weights

#Correlation of predictions with true values in test set
uhatmat<-matrix(funout$uhat, ncol=3)
uhatvec<-rowSums(uhatmat)
cor(Y[-Trainsamp], uhatvec[-Trainsamp])

---

**emmremlMultivariate**

*Function to fit multivariate Gaussian mixed model with known covariance structure.*

**Description**

This function estimates the parameters of the model

\[ Y = BX + GZ + E \]

where \( Y \) is the \( dxn \) matrix of response variable, \( X \) is a \( qxn \) known design matrix of fixed effects, \( Z \) is a \( lxn \) known design matrix of random effects, \( B \) is \( dxq \) matrix of fixed effects coefficients and \( G \) and \( E \) are independent matrix variate variables with \( N_{dxl}(0, V_G, K) \) and \( N_{dxn}(0, V_E, I_n) \) correspondingly. It also produces the BLUPs for the random effects \( G \) and some other statistics.

**Usage**

```r
emmremlMultivariate(Y, X, Z, K, varBhat=FALSE, varGhat=FALSE, PEVWhat=FALSE, test=FALSE, tolpar=1e-06, tolparinv=1e-06)
```

**Arguments**

- **Y** \( dxn \) matrix of response variable
- **X** \( qxn \) known design matrix of fixed effects
- **Z** \( lxn \) known design matrix of random effects
- **K** \( lx1 \) matrix of known relationships
- **varBhat** TRUE or FALSE
- **varGhat** TRUE or FALSE
- **PEVWhat** TRUE or FALSE
- **test** TRUE or FALSE
- **tolpar** tolerance parameter for convergence
- **tolparinv** tolerance parameter for matrix inverse
**Value**

- \( V_g \): Estimate of \( V_G \)
- \( V_e \): Estimate of \( V_E \)
- \( \hat{b} \): BLUEs for \( B \)
- \( \hat{g} \): BLUPs for \( G \)
- \( \chi^2 \): Test statistics for testing whether the fixed effect coefficients are equal to zero.
- \( p values \): Pvalues obtained from large sample theory for the fixed effects. We report the pvalues adjusted by the "padjust" function for all fixed effect coefficients.
- \( \chi^2 \): Test statistic values for testing whether the BLUPs are equal to zero.
- \( \hat{V}_g \): Large sample variance for BLUPs.
- \( \hat{V}_b \): Large sample variance for the elements of \( B \).
- \( PEV_{\hat{g}} \): Prediction error variance estimates for the BLUPs.

**Examples**

```r
l <- 20
m <- 15
m <- 40

M <- matrix(rbinom(m * l, 1, 2), nrow = l)
rownames(M) <- paste("l", 1:nrow(M))
beta1 <- rnorm(m) * exp(rbinom(m, 5, 0.2))
beta2 <- rnorm(m) * exp(rbinom(m, 5, 1))
beta3 <- rnorm(m) * exp(rbinom(m, 5, 1)) + beta2

g1 <- M * beta1
g2 <- M * beta2
g3 <- M * beta3
e1 <- sd(g1) * rnorm(l)
e2 <- (-e1 * 2 * sd(g2) / sd(g1) + .25 * sd(g2) / sd(g1)) * rnorm(l)
e3 <- (e1 + .25 * sd(g2) / sd(g1) + .25 * sd(g2) / sd(g1)) * rnorm(l)

y1 <- -10 + g1 + e1
y2 <- -50 + g2 + e2
y3 <- -5 + g3 + e3

Y <- rbind(t(y1), t(y2), t(y3))
rownames(Y) <- rownames(M)
cov(t(Y))
Y[1:3, 1:5]

K <- cov(t(M))
K <- K / mean(diag(K))
rownames(K) <- colnames(K) <- rownames(M)
X <- matrix(1, nrow = 1, ncol = 1)
```
colnames(X) <- rownames(M)
Z <- diag(1)
rownames(Z) <- colnames(Z) <- rownames(M)
SampleTrain <- sample(rownames(Z), n)
Ztrain <- Z[rownames(Z) %in% SampleTrain,]
Ztest <- Z[!(rownames(Z) %in% SampleTrain),]

## For a quick answer, tolpar is set to 1e-4. Correct this in practice.
outfunc <- emmremlMultivariate(Y = Y %*% t(Ztrain),
X = X %*% t(Ztrain), Z = t(Ztrain),
K = K, tolpar = 1e-4, varBhat = FALSE,
varGhat = FALSE, PEVGhat = FALSE, test = FALSE)
Yhatest <- outfunc$Gpred %*% t(Ztest)

cor(cbind(Ztest %*% Y[1,], Ztest %*% outfunc$Gpred[1,],

outfuncRidgeReg <- emmremlMultivariate(Y = Y %*% t(Ztrain), X = X %*% t(Ztrain), Z = t(Ztrain %*% M),
K = diag(m), tolpar = 1e-5, varBhat = FALSE, varGhat = FALSE,
PEVGhat = FALSE, test = FALSE)
Gpred2 <- outfuncRidgeReg$Gpred %*% t(M)
cor(Ztest %*% Y[1,], Ztest %*% Gpred2[1,])
cor(Ztest %*% Y[2,], Ztest %*% Gpred2[2,])
cor(Ztest %*% Y[3,], Ztest %*% Gpred2[3,])
Index

EMMREML, 2
emmreml, 2
emmremlMultiKernel, 4
emmremlMultivariate, 6