# Package ‘MM4LMM’

January 15, 2019

**Type** Package

**Title** Inference of Linear Mixed Models Through MM Algorithm

**Version** 1.1.5

**Date** 2019-01-15

**Author** Fabien Laporte, Tristan Mary-Huard

**Maintainer** Fabien Laporte <fabien.laporte@inra.fr>

**Description** The main function MMEst() performs (Restricted) Maximum Likelihood in a variance component mixed models using a Min-Max (MM) algorithm (Hunter, D. R., & Lange, K. (2004) <doi:10.1198/0003130042836>).

**License** GPL (>= 2)

**Imports** Rcpp (>= 0.12.13), Matrix, parallel, stats

**LinkingTo** Rcpp, RcppEigen

**NeedsCompilation** yes

**Depends** R (>= 2.10)

**Repository** CRAN

**Date/Publication** 2019-01-15 10:50:06 UTC

## R topics documented:

MM4LMM-package *Min-Max algorithms for Variance Component Mixed Model Inference.*

## Description

This package provides a function to perform either ML or ReML estimation in a Variance Component Mixed Model. The optimization of the (possibly Restricted) log-likelihood is perfomed using the Min-Max algorithm described in Hunter et al. (2004). Depending on the number of variance components, different computational tricks are used to speed up inference. Additionally, the AnovaTest function provides Type I and Type III test series for fixed effects. The nullity of a specific linear combination can also be tested.

## Details

Variance Component Mixed Models are mixed models of the form

$$Y = X\beta + \sum_{k=1}^{K} Z_k u_k$$

where Y is the response vector, X and $\beta$ are respectively the incidence matrix and the coefficient vector associated with the fixed effects, $u_k$ is the kth vector of random effects and corresponds to its associated incidence matrix. All random effect are assumed to be Gaussian with mean 0 and covariance $\sigma_k^2 R_k$, where $R_k$ is a known correlation matrix and $\sigma_k^2$ is an unknown variance parameter. All random effects are assumed to be independent. In many applications the last random component corresponds to the error and therefore both $Z_k$ and $R_k$ correspond to the identity matrix.

In such models the inference of both the unknown variance components $\sigma_k^2$,...,$\sigma_K^2$ and the fixed effect $\beta$ can be achieved through Maximum Likelihood (ML) or Restricted Maximum Likelihood (ReML) estimation. Neither ML nor ReML yield close form expressions of the estimates, consequently the maximization of the (restricted) log-likelihood has to be performed numerically. This package provides the user with Min-Max algorithms for the optimization. Efficient tricks such as profiling, MME trick and MM acceleration are implemented for computational efficiency (see Johnson et al. (1995), Varadhan et al. (2008) for details). The main function MMEst handles parallel inference of multiple models sharing the same set of correlation matrices - but possibly different fixed effects, an usual situation in GWAS analysis for instance.

## Author(s)

Fabien Laporte and Tristan Mary-Huard

Maintainer: Fabien Laporte <fabien.laporte@inra.fr>

## References

Johnson, D. L., & Thompson, R. (1995). Restricted maximum likelihood estimation of variance components for univariate animal models using sparse matrix techniques and average information. Journal of dairy science, 78(2), 449-456.

Hunter, D. R., & Lange, K. (2004). A tutorial on MM algorithms. The American Statistician, 58(1), 30-37.

Varadhan, R., & Roland, C. (2008). Simple and globally convergent methods for accelerating the convergence of any EM algorithm. Scandinavian Journal of Statistics, 35(2), 335-353.

Zhou, H., Hu, L., Zhou, J., & Lange, K. (2015). MM algorithms for variance components models. arXiv preprint arXiv:1509.07426.

---

`AnovaTest` *Type I and Type III Tests for mixed models.*

---

### Description

This function computes Type I and Type III tests for each fixed effect of a model, as displayed by the `MMEst` function. Alternatively, a specific linear combination of the fixed parameters may be tested (at 0).

### Usage

```
AnovaTest(ResMMEst , TestedCombination=NULL , Type = "TypeIII" , NbCores=1)
```

### Arguments

`ResMMEst` A list as displayed by the `MMEst` function.

`TestedCombination` A contrast matrix or a list of contrast matrices $C_m$. Each matrix corresponds to a (set of) linear combination to be (jointly) tested at 0.

`Type` Either "TypeI" of "TypeIII" (default is "TypeIII"). AnovaTest will compute tests of the given type for each fixed effect in the model. The option is ignored if a `TestedCombination` is provided.

`NbCores` The number of cores to be used.

### Details

If no `TestedCombination` is provided, the function performs either Type I or Type III tests for each fixed effect in the model (default is Type III). If `TestedCombination` is provided, each linear combination in `TestedCombination` is tested at 0 using a Wald test. No check is performed regarding the estimability of the linear combination to be tested. If the dimension of the combination does not match with the dimension of `ResMMEst`, a `NA` is returned.

### Value

The output of the function is a list with as many items as in the original input list `ResMMEst`. For each item of `ResMMEst`, a table is provided that contains the Wald test statistics, p-values and degrees of freedom for all tested hypotheses.

### Author(s)

F. Laporte and T. Mary-Huard

## Examples

```
require('MM4LMM')
data(QTLDetectionExample)
Pheno <- QTLDetectionExample$Phenotype
Geno <- QTLDetectionExample$Genotype
Kinship <- QTLDetectionExample$Kinship

##Build the VarList object
VL <- list(Additive = Kinship , Error = diag(1,length(Pheno)))

##Perform inference
Result <- MMEst(Y=Pheno , X = Geno , VarList = VL)

##Compute tests
AOV <- AnovaTest(Result,Type="TypeI")

##Test specific contrast matrix
TC = matrix(c(0,1),nrow=1)
AOV2 <- AnovaTest(Result, TestedCombination = TC)
```

---

# MMEst — *MM inference method for variance component mixed models*

---

## Description

This is the main function of the *MM4LMM* package. It performs inference in a variance component mixed model using a Min-Max algorithm. Inference in multiple models (e.g. for GWAS analysis) can also be performed.

## Usage

```
MMEst(Y, Cofactor = NULL, X = NULL, formula=NULL, VarList, ZList = NULL, Method = "Reml",
Henderson=NULL, Init = NULL, CritVar = 0.001, CritLogLik = 0.001,
MaxIter = 100, NbCores = 1)
```

## Arguments

| | |
|---|---|
| `Y` | A vector of response values. |
| `Cofactor` | An incidence matrix corresponding to fixed effects common to all models to be adjusted. If `NULL`, a single intercept is used as cofactor. |
| `X` | An incidence matrix or a list of incidence matrices corresponding to fixed effects specific to each model. If `X` is a matrix, one model per column will be fitted. If `X` is a list, one model per element of the list will be fitted (default is `NULL`). |
| `formula` | A formula object specifying the fixed effect part of all models separated by + operators. To specify an interaction between `Cofactor` and `X` use the colnames of `X` when it is a list or use "Xeffect" when `X` is a matrix. |
| `VarList` | A list of correlation matrices associated with random and residual effects. |

| | |
|---|---|
| `ZList` | A list of incidence matrices associated with random and residual effects (default is `NULL`). |
| `Method` | The method used for inference. Available methods are "Reml" (Restricted Maximum Likelihood) and "ML" (Maximum Likelihood). |
| `Henderson` | If `TRUE` the Henderson trick is applied. If `FALSE` the Henderson trick is not applied. If `NULL` the algorithm chooses wether to use the trick or not. |
| `Init` | A vector of initial values for variance parameters (default is `NULL`) |
| `CritVar` | Value of the criterion for the variance components to stop iteration. (see Details) |
| `CritLogLik` | Value of the criterion for the log-likelihood to stop iteration. (see Details) |
| `MaxIter` | Maximum number of iterations per model. |
| `NbCores` | Number of cores to be used. |

## Details

If `X` is `NULL`, the following model is fitted:

$$Y = X_C\beta_C + \sum_{k=1}^{K} Z_k u_k$$

with $X_C$ the matrix provided in `Cofactor`, $\beta_C$ the unknown fixed effects, $Z_k$ the incidence matrix provided for the kth component of `ZList` and $u_k$ the kth vector of random effects. If `ZList` is unspecified, all incidence matrices are assumed to be the Identity matrix. Random effects are assumed to follow a Gaussian distribution with mean 0 and covariance matrix $R_k\sigma_k^2$, where $R_k$ is the kth correlation matrix provided in `VarList`.

If `X` is not `NULL`, the following model is fitted for each i:

$$Y = X_C\beta_C + X_{[i]}\beta_{[i]} + \sum_{k=1}^{K} Z_k u_k$$

where $X_{[i]}$ is the incidence matrix corresponding to the ith component (i.e. column if $X$ is a matrix, element otherwise) of $X$, and $\beta_{[i]}$ is the (unknow) fixed effect associated to $X_{[i]}$.

All models are fitted using the MM algorithm. If `Henderson=TRUE`, at each step the quantities required for updating the variance components are computed using the Mixed Model Equation (MME) trick. See Johnson et al. (1995) for details.

## Value

The result is a list where each element corresponds to a fitted model. Each element displays the following:

| | |
|---|---|
| `Beta` | Estimated values of $\beta_C$ and $\beta_i$ |
| `Sigma2` | Estimated values of $\sigma_1^2, ..., \sigma_K^2$ |
| `VarBeta` | Variance matrix of `Beta` |
| `LogLik (Method)` | The value of the (restricted, if `Method` is "Reml") log-likelihood |

| | |
|---|---|
| NbIt | The number of iterations required to reach the optimum |
| Method | The method used for the inference |
| attr | An integer vector with an entry for each element of Beta giving the term in Factors which gave rise to this element (for internal use in the function AnovaTest) |
| Factors | Names of each term in the formula |

## Author(s)

F. Laporte and T. Mary-Huard

## References

Johnson, D. L., & Thompson, R. (1995). Restricted maximum likelihood estimation of variance components for univariate animal models using sparse matrix techniques and average information. Journal of dairy science, 78(2), 449-456.

Hunter, D. R., & Lange, K. (2004). A tutorial on MM algorithms. The American Statistician, 58(1), 30-37.

Zhou, H., Hu, L., Zhou, J., & Lange, K. (2015). MM algorithms for variance components models. arXiv preprint arXiv:1509.07426.

## Examples

```
require('MM4LMM')

#### Example 1: variance component analysis, 1 model
data(VarianceComponentExample)
DataHybrid <- VarianceComponentExample$Data
KinF <- VarianceComponentExample$KinshipF
KinD <- VarianceComponentExample$KinshipD

##Build incidence matrix for each random effect
Zf <- t(sapply(as.character(DataHybrid$CodeFlint), function(x)
  as.numeric(rownames(KinF)==x)))
Zd <- t(sapply(as.character(DataHybrid$CodeDent), function(x)
  as.numeric(rownames(KinD)==x)))

##Build the VarList and ZList objects
VL = list(Flint=KinF , Dent=KinD , Error = diag(1,nrow(DataHybrid)))
ZL <- list(Flint=Zf , Dent=Zd , Error = diag(1,nrow(DataHybrid)))

##Perform inference
#A first way to call MMEst
ResultVA <- MMEst(Y=DataHybrid$Trait  , Cofactor = matrix(DataHybrid$Trial)
                  , ZList = ZL  ,  VarList = VL)
length(ResultVA)
print(ResultVA)

#A second way to call MMEst (same result)
Formula <- as.formula('~ Trial')
ResultVA2 <- MMEst(Y=DataHybrid$Trait  , Cofactor = DataHybrid,
```

```
                  formula = Formula
                 , ZList = ZL  ,  VarList = VL)
length(ResultVA2)
print(ResultVA2)



#### Example 2: Marker Selection with interaction between Cofactor and X matrix
Formula <- as.formula('~ Trial+Xeffect+Xeffect:Trial')
ResultVA3 <- MMEst(Y=DataHybrid$Trait  , Cofactor = DataHybrid,
                  X = VarianceComponentExample$Markers,
                  formula = Formula
                 , ZList = ZL  ,  VarList = VL)
length(ResultVA3)
print(ResultVA3[[1]])


#### Example 3: QTL detection with two variance components
data(QTLDetectionExample)
Pheno <- QTLDetectionExample$Phenotype
Geno <- QTLDetectionExample$Genotype
Kinship <- QTLDetectionExample$Kinship

##Build the VarList object
VLgd <- list(Additive=Kinship , Error=diag(1,length(Pheno)))

##Perform inference
ResultGD <- MMEst(Y=Pheno , X=Geno
                 , VarList=VLgd , CritVar = 10e-5)

length(ResultGD)
print(ResultGD[[1]])
```

---

QTLDetectionExample *QTL Detection Example*

---

## Description

This corresponds to (a sample of) the dataset presented in Rincent et al. (2014).

## Usage

```
data("QTLDetectionExample")
```

## Format

The format is: List of 3

`Phenotype` Named num [1:259]

`Genotype` int [1:259,1:10]

`Kinship` num [1:259,1:259]

## Details

The list contains three elements:

- Phenotype: a numeric vector containing phenotypes of individuals
- Genotype: a matrix containing the genotypes of indviduals over 10 biallelic markers
- Kinship: a matrix of simple relatedness coefficients between individuals

## Source

https://link.springer.com/article/10.1007%2Fs00122-014-2379-7

## References

Rincent, R., Nicolas, S., Bouchet, S., Altmann, T., Brunel, D., Revilla, P., ... & Schipprack, W. (2014). Dent and Flint maize diversity panels reveal important genetic potential for increasing biomass production. Theoretical and applied genetics, 127(11), 2313-2331.

## Examples

```
data(QTLDetectionExample)
names(QTLDetectionExample)
## maybe str(QTLDetectionExample) ; plot(QTLDetectionExample) ...
```

---

VarianceComponentExample

# *Variance Component Example*

---

## Description

This corresponds to (a sample of) the dataset presented in Giraud et al. (2017).

## Usage

```
data("VarianceComponentExample")
```

## Format

The format is: List of 3

`Data` 'data.frame': 432 obs. of 5 variables

- `Trial` a factor with 2 levels
- `CodeHybrid` a factor with 177 levels
- `CodeDent` a factor with 116 levels
- `CodeFlint` a factor with 122 levels
- `Trait` a numeric vector

`KinshipD` num [1:116,1:116]

`KinshipF` num [1:122,1:122]

## Details

The list contains three elements:

- Data: a data frame containing the information about hybrids (trials, hybrid names, dent parental lines, flint parental lines and phenotypes)
- KinshipD: a matrix of simple relatedness coefficients between dent lines
- KinshipF: a matrix of simple relatedness coefficients between flint lines

## Source

http://www.genetics.org/content/207/3/1167.figures-only

## References

Giraud, H., Bauland, C., Falque, M., Madur, D., Combes, V., Jamin, P., ... & Blanchard, P. (2017). Reciprocal Genetics: Identifying QTL for General and Specific Combining Abilities in Hybrids Between Multiparental Populations from Two Maize (Zea mays L.) Heterotic Groups. Genetics, 207(3), 1167-1180.

## Examples

```
data(VarianceComponentExample)
names(VarianceComponentExample)
## maybe str(VarianceComponentExample) ; plot(VarianceComponentExample) ...
```

# Index