Package ‘rlme’

Type Package
Title Rank-Based Estimation and Prediction in Random Effects Nested Models
Version 0.5
Date 2018-01-08
Author Yusuf Bilgic, Herb Susmann and Joseph McKean
Maintainer Yusuf Bilgic <bilgic@geneseo.edu>
License GPL (>= 2)
Imports MASS, quantreg, nlme, mgcv, stringr, magic, robustbase, Rcpp, stats, utils, graphics
Suggests testthat
NeedsCompilation yes
LinkingTo Rcpp
Repository CRAN
RoxygenNote 6.0.1
LazyData true
Date/Publication 2018-01-09 17:35:55 UTC

R topics documented:

   rlme-package .......................................................... 2
   beta_var .............................................................. 3
   compare.fits ....................................................... 4
   dispvar .............................................................. 5
   fitdvcov .............................................................. 6
   GEER_est ............................................................. 7
   getgrstplot ......................................................... 8
rlme-package

Description

An R package for rank-based robust estimation and prediction in random effects nested models

Details

Package: rlme
Type: Package
Version: 0.2
Date: 2013-07-07
License: GPL (>= 2)

Author(s)

Yusuf Bilgic <bilgic@geneseo.edu>, Herb Susmann <hps1@geneseo.edu> and Joseph McKean <joemckean@yahoo.com>

Maintainer: Yusuf Bilgic <bilgic@geneseo.edu> or <yusuf.k.bilgic@gmail.com>
See Also

rlme

Examples

```r
library(rlme)
data(schools)
formula = y ~ 1 + sex + age + (1 | region) + (1 | region:school)
rlme.fit = rlme(formula, schools)
summary(rlme.fit)
```

---

**beta_var**

Estimate fixed-effect variance for Joint Rank Method (JR) in three-level nested design.

**Description**

Fixed effect variance estimation for Joint Rank Method (JR). It assumes Compound Symmetric (CS) structure of error terms. For k-level design, there are k-1 intra/inter-class parameters to place in a correlation matrix of errors.

**Usage**

`beta_var(x, school, tauhat, v1, v2, v3, section, mat)`

**Arguments**

- `x`: Data frame of covariates.
- `school`: A vector of cluster.
- `tauhat`: This is obtained from Rank-based fitting. `tauhat` here~~
- `v1`: This is 1, main diagonal element for correlation matrix of observations. Correlation of an observation with itself is 1.
- `v2`: Intra-cluster correlation coefficient.
- `v3`: Intra-subcluster correlation coefficient.
- `section`: A vector of subclusters, nx1.
- `mat`: A matrix of numbers of observations in subclusters. Dimension is Ixmax(number of subclusters). Each row indicates one cluster.

**Details**

Correlation coefficients are obtained using Moment Estimates. See Klole et al (2009), Bilgic (2012) and HM (2012)
**Value**

*var*  
The variance of fixed estimated.

**Author(s)**

Yusuf Bilgic

**References**


---

**compare.fits**  
*Compare Fits*

**Description**

Comparer two model fits. It returns tdbeta value and cfits values of two fits. The function uses the fixed effects estimates from fit 1 and fit 2 along with the covariance of the rank-based fit.

**Usage**

```r
compare.fits(x, fit1, fit2)
```

**Arguments**

- **x**  
  Matrix of covariates
- **fit1**  
  A class of type rlme.
- **fit2**  
  A class of type rlme.

**Value**

Returns tdbeta and cfits values.

**See Also**

`fitdvcov`
dispvar

Examples

data(schools)
model = y ~ 1 + sex + age + (1 | region) + (1 | region:school)

# Extract covariants into matrix
cov = as.matrix(data.frame(schools[,"sex"], schools[,"age"]))

# Fit the models using each method
reml.fit = rlme(model, schools, method="reml")
gr.fit = rlme(model, schools, method="gr")

compare.fits(cov, reml.fit, gr.fit)

dispvar

Rank-based dispersion estimate.

Description

This is an unbiased estimator with a correction factor for standard deviation when normal errors.

Usage

dispvar(x, score = 1)

Arguments

x vector
score score type - 1 or 2

References

Description
Obtains measurement for the fits based on estimates beta1, beta2 and covariance matrix from a rank based methods.

Usage
fitdvcov(x1, beta1, beta2, vcw)

Arguments
x1 data
beta1 model 1 beta estimate
beta2 model 2 beta estimate
vcw variance matrix

See Also
compare.fits

Examples

# Compare GR and JR methods
data(schools)
model = y ~ 1 + sex + age + (1 | region) + (1 | region:school)
# Extract covariants into matrix
cov = as.matrix(data.frame(schools[,"sex"], schools[,"age"]))
# Fit the models using each method
jr.fit = rlme(model, schools, method="jr")
gr.fit = rlme(model, schools, method="gr")
# Extract beta estimates, ignoring the intercept
jr.beta = jr.fit$fixed.effects$Estimate[c(2, 3)]
gr.beta = gr.fit$fixed.effects$Estimate[c(2, 3)]
# Extract beta variance matrix
var.b = jr.fit$var.b
fitdvcov(cov, jr.beta, gr.beta, var.b)
GEER_est

GEER: General Estimating Equation Rank-Based Estimation Method

Description

The package rlme calls this function for gee method, one of the methods proposed in Bilgic’s study (2012). Also see Kloke et al. (2013). concise (1-5 lines) description of what the function does. ~

Usage

GEER_est(x, y, I, sec, mat, school, section, weight = "wil", rprpair = "hl-disp", verbose = FALSE)

Arguments

x Design matrix, pxn, without intercept.
y Response vector of nx1.
I Number of clusters.
sec A vector of subcluster numbers in clusters.
mat A matrix of numbers of observations in subclusters. Dimension is lmxmax(number ofsubclusters). Each row indicates one cluster.
school A vector of clusters, nx1.
section A vector of subclusters, nx1.
weight When weight="hbr", it uses hbr weights in GEE weights. By default, ="wil", it uses Wilcoxon weights. See the theory in the references.
rprpair By default, it uses "hl-disp" in the random prediction procedure (RPP). Also, "med-mad" would be an alternative.
verbose Boolean indicating whether to print out diagnostic messages.

Value

theta Fixed effect estimates.
ses Standard error for the fixed estimates.
sigma Variances of cluster, subcluster, and residual.
ehat Raw error.
ehats Independence error from last weighted step.
effect_sch Cluster random error.
effect_sec Subcluster random error.
effect_err Epsilon error.

Author(s)

Yusuf K. Bilgic, yekabe@hotmail.com
References


See Also

rlme, GR_est, JR_est, rprmeddisp

Examples

# See the rlme function.

getgrstplot

Q-Q Plot and Standardized Residual Plot for the GR fit.

Description

It gets Q-Q Plot and Standardized Residual Plot of residuals.

Usage

getgrstplot(rlme.fit)

Arguments

rlme.fit RLME fit object

Details

The fit is obtained from rlme()

See Also

rlme


getlmestplot

Q-Q Plot and Standardized Residual Plot for the REML or ML fit.

Description

It gets Q-Q Plot and Standardized Residual Plot of residuals. concise (1-5 lines) description of what the function does.

Usage

getlmestplot(rlme.fit)

Arguments

rlme.fit The fit is obtained from rlme()

See Also

rlme

GR_est

GR Method

Description

Fits a model using the GR method

Usage

GR_est(x, y, I, sec, mat, school, section, rprpair = "hl-disp", verbose = FALSE)

Arguments

x Covariate matrix or data frame.
y Response matrix or data frame.
I Number of clusters
sec A vector of subcluster numbers in clusters.
mat A matrix of numbers of observations in subclusters. Dimension is Ixmax(number of subclusters). Each row indicates one cluster.
school A vector of clusters, nx1.
section A vector of subclusters, nx1.
rprpair By default, it uses "hl-disp" in the random prediction procedure (RPP). Also, "med-mad" would be an alternative.
verbose Boolean indicating whether to print out messages from the algorithm.
Value

- **theta**: Fixed effect estimates.
- **ses**: Standard error for the fixed estimates.
- **sigma**: Variances of cluster, subcluster, and residual.
- **ehat**: Raw error.
- **ehats**: Independence error from last weighted step.
- **effect_sch**: Cluster random error.
- **effect_sec**: Subcluster random error.
- **effect_err**: Epsilon error.

Author(s)

Yusuf Bilgic

Examples

# See rlme function

```
hbrwts_gr
```

### Description

Calculates hbr weights for the GEER method. This turns a vector of weights for a vector of errors. Used to make factor space more robust, up to 50% breakdown. See HM (2012) and Terpstra and McKean (2005) for details. The ww package produces this weights as well.

#### Usage

```
hbrwts_gr(xmat, y, percent = 0.95, intest = ltsreg(xmat, y)$coef)
```

#### Arguments

- **xmat**: Design matrix, pxn, without intercept.
- **y**: Response vector in nx1.
- **percent**: This is 0.95.
- **intest**: This is obtained from myltsreg(xmat, y)$coef

#### Details

The ww package explains how it is obtained.
**Instruction**

**Author(s)**

J. W. McKean

**References**


**See Also**

GEER_est

---

**Description**

A data frame on school instruction results.

**Format**

A data frame with 1190 observations on the following 13 variables.

- **X** a numeric vector
- **girl** a numeric vector
- **minority** a numeric vector
- **mathkind** a numeric vector
- **mathgain** a numeric vector
- **ses** a numeric vector
- **yearstea** a numeric vector
- **mathknow** a numeric vector
- **housepov** a numeric vector
- **mathprep** a numeric vector
- **classid** a numeric vector identifying the class within school
- **schoolid** a numeric vector identifying the school
- **childid** a numeric vector

**Source**

Examples

# The following code takes a few minutes to run.
# In the interest of saving CRAN's example testing time,
# it has been commented out. If you want to use it,
# just uncomment and run.

# data(instruction)
# attach(instruction)

# data = data.frame(
#  y = mathgain,
#  mathkind = mathkind,
#  girl = girl,
#  minority = minority,
#  ses = ses,
#  school = factor(schoolid),
#  section = factor(classid))

# fit.rlme = rlme(y ~ 1 + mathkind + girl + minority + ses + (1 | school) + (1 | school:section),
#  data = data,
#  method = "gr")

# summary(fit.rlme)

---

JR_est  

JR Method

Description

Fit a model using the JR method

Usage

JR_est(x, y, I, sec, mat, school, section, rprpair = "hl-disp",
       verbose = FALSE)

Arguments

x  Covariate matrix or data frame
y  Response matrix or data frame
I  Number of clusters.
sec  A vector of subcluster numbers in clusters.
mat  A matrix of numbers of observations in subclusters. Dimension is Ix\(\text{max}\)(number
    of subclusters). Each row indicates one cluster. mat here--
school  A vector of clusters, nx1.
section A vector of subclusters, nx1.

rprpair By default, it uses "hl-disp" in the random prediction procedure (RPP). Also, "med-mad" would be an alternative.

verbose Boolean indicating whether to print out diagnostic messages.

Value

theta Fixed effect estimates.

ses Standard error for the fixed estimates.

sigma Covariate variance estimates using RPP (Groggel and Dubnicka’s procedure).

ehat Raw error.

effect_sch Cluster random error.

effect_sec Subcluster random error.

effect_err Epsilon error.

Author(s)

Yusuf Bilgic

See Also

rlme

lmr (Rank Based Fixed Effect Regression)

Description

Computes rank based regression estimates for fixed effect models.

Usage

lmr(f, data, se = FALSE, method = "L-BFGS-B")

Arguments

f A model formula

data Data to use for model fitting

se Boolean indicating whether or not to calculate standard errors for intercept and slope estimates

method Optimization method to use. Will accept any method usable by optim, e.g. one of c("Nelder-Mead", "BFGS", "CG", "L-BFGS-B", "SANN", "Brent"). "BFGS" or "L-BFGS-B" are recommended. "L-BFGS-B" should be used for large datasets to conserve memory.
Value

<table>
<thead>
<tr>
<th>fixed.effects</th>
<th>Fixed effect estimates</th>
</tr>
</thead>
<tbody>
<tr>
<td>ehat</td>
<td>Residuals from model</td>
</tr>
</tbody>
</table>

Author(s)

Herb Susmann

See Also

rlme, optim

Examples

```
# load schools data
data(schools)

# Fit fixed effects model with lmr
lmr.fit = lmr(y ~ age + sex, data=schools)
summary(lmr.fit)

# Fit with lmr and calculate standard errors
lmr.fit = lmr(y ~ age + sex, data=schools, se=TRUE)
summary(lmr.fit)
```

---

**LM_est**  
Linear Model Estimation using the nlme package.

Description

This gets the REML or ML estimates and predictions of random effects from the nlme package.

Usage

```
LM_est(x, y, dat, method = "REML")
```

Arguments

- **x**: Design matrix, (p+1)xn, with intercept.
- **y**: Response vector of nx1.
- **dat**: Data frame
- **method**: Character string indicating method to use, either "ML" or "REML" (defaults to REML).
Value

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>theta</td>
<td>Fixed effects estimates.</td>
</tr>
<tr>
<td>ses</td>
<td>Standard error for fixed effects.</td>
</tr>
<tr>
<td>varb</td>
<td>Variances.</td>
</tr>
<tr>
<td>sigma</td>
<td>Error.</td>
</tr>
<tr>
<td>ehat</td>
<td>Raw residuals</td>
</tr>
<tr>
<td>standr.lme</td>
<td>Standardized residual</td>
</tr>
<tr>
<td>effect_sch</td>
<td>Cluster random error.</td>
</tr>
<tr>
<td>effect_sec</td>
<td>Subcluster random error.</td>
</tr>
<tr>
<td>effect_err</td>
<td>Epsilon error.</td>
</tr>
</tbody>
</table>

Author(s)

Yusuf Bilgic

References


See Also

rlme

Description

Uses optim to find regression estimates which minimize dispersion function on X and Y input matrices

Usage

minimize_dispersion(X, Y, method = "BFGS", init.guess = "quantreg", verbose = FALSE, se = TRUE)
**plot.rlme**

### Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>Input matrix</td>
</tr>
<tr>
<td>Y</td>
<td>Response vector</td>
</tr>
<tr>
<td>method</td>
<td>Method optim should use - one of &quot;Nelder-Mead&quot;, &quot;BFGS&quot;, &quot;CG&quot;, &quot;L-BFGSB&quot;, &quot;SANN&quot;, or &quot;Brent&quot;.</td>
</tr>
<tr>
<td>init.guess</td>
<td>How to calculate the first regression estimate. Defaults to using quantile regression.</td>
</tr>
<tr>
<td>verbose</td>
<td>Whether to print out verbose messages.</td>
</tr>
<tr>
<td>se</td>
<td>Whether or not to calculate standard errors of regression estimates.</td>
</tr>
</tbody>
</table>

### Value

<table>
<thead>
<tr>
<th>Object</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>theta</td>
<td>Regression parameter estimates</td>
</tr>
<tr>
<td>ehat</td>
<td>Regression residuals</td>
</tr>
</tbody>
</table>

### Author(s)

Herb Susmann

---

**plot.rlme**

*Plot rlme Fit*

---

**Description**

Generates Normal Q-Q plot of residuals from rlme fit

**Usage**

```r
## S3 method for class 'rlme'
plot(x, ...)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
<td>A list of class rlme. Store as fit.rlme.</td>
</tr>
<tr>
<td>...</td>
<td>not used</td>
</tr>
</tbody>
</table>

**Examples**

```r
data(schools)
rlme.fit = rlme(y ~ 1 + sex + age + (1 | region) + (1 | region:school), schools, method="gr")
plot(rlme.fit)
```
rhosch

Cluster Correlation Coefficient Estimate

Description

Moment estimate version of correlation coefficient in a cluster in a three-level nested design.

Usage

rhosch(ahat, school, section)

Arguments

ahat  A vector of scores. Wilcoxon scores are used in the package.
school  A vector of clusters.
section  A vector of subclusters.

References


rhosect

Subcluster Correlation Coefficient Estimate

Description

Moment estimate version of correlation coefficient in a subcluster in a three-level nested design.

Usage

rhosect(ahat, school, section)

Arguments

ahat  A vector of scores. Wilcoxon scores are used in the package.
school  A vector of clusters.
section  A vector of subclusters.

References

**Description**

This function estimates fixed effects and predicts random effects in two- and three-level random effects nested models using three rank-based fittings (GR, GEER, JR) via the prediction method algorithm RPP.

**Usage**

```r
rlme(f, data, method = "gr", print = FALSE, na.omit = TRUE, weight = "wil", rprpair = "hl-disp", verbose = FALSE)
```

**Arguments**

- `f` : An object of class formula describing the mixed effects model. The syntax is same as in the lme4 package. Example: `y ~ 1 + sex + age + (1 | region) + (1 | region:school)` - sex and age are the fixed effects, region and school are the nested random effects, school is nested within region.
- `data` : The dataframe to analyze. Data should be cleaned prior to analysis: cluster and subcluster columns are expected to be integers and in order (e.g. all clusters and subclusters).
- `method` : string indicating the method to use (one of "gr", "jr", "reml", and "geer"). defaults to "gr".
- `print` : Whether or not to print a summary of results. Defaults to false.
- `na.omit` : Whether or not to omit rows containing NA values. Defaults to true.
- `weight` : When weight="bbr", it uses bbr weights in GEE weights. By default, ="wil", it uses Wilcoxon weights. See the theory in the references.
- `rprpair` : By default, it uses "hl-disp" in the random prediction procedure (RPP). Also, "med-mad" would be an alternative.
- `verbose` : Boolean indicating whether to print out diagnostic messages.

**Details**

The iterative methods GR and GEER can be quite slow for large datasets; try JR for faster analysis. If you want to use the GR method, try using rprpair='med-mad'. This method avoids building a NxN covariance matrix which can quickly become unwieldy with large data.

**Value**

The function returns a list of class "rlme". Use summary.rlme to see a summary of the fit.

- `formula` : The model formula.
- `method` : The method used.
fixed.effects  Estimate of fixed effects.
random.effects  Estimate of random effects.
standard.residual  Residuals.
intra.class.correlations  Intra/inter-class correlation estimates obtained from RPP.
t.value  t-values.
p.value  p-values.
location  Location.
scale  Scale.
y  The response variable y.
num.obs  Number of observations in provided dataset.
num.clusters  The number of clusters.
num.subclusters  The number of subclusters.
effect.err  Effect from error.
effect.cluster  Effect from cluster.
effect.subcluster  Effect from subcluster.
var.b  Variances of fixed effects estimate (Beta estimates).
xstar  Weighted design matrix with error covariance matrix.
ystar  Weighted response vector with its covariance matrix.
ehat  The raw residual.
ehats  The raw residual after weighted step. Scaled residual.

Author(s)
Yusuf Bilgic <yekabe@hotmail.com> and Herb Susmann <hps1@geneseo.edu>

References

See Also
summary.rlme, plot.rlme, compare.fits
Examples

    data(schools)
    rlme.fit = rlme(y ~ 1 + sex + age + (1 | region) + (1 | region:school), schools, method="gr")
    summary(rlme.fit)

    # Try method="geer", "reml", "ml" and "jr" along with
    # rprpair="hl-disp" (not robust), and "med-mad" (robust),
    # weight="hbr" is for the gee method.

rpr  Cluster and Subcluster effects

Description

    Partitions model residuals into cluster and subcluster effects using RPP algorithm.

Usage

    rpr(f, resid, data, rprpair = "hl-disp")

Arguments

    f  A model formula which specifies the random effects (see example)
    resid The residuals from the fitted model
    data The data the model was fitted on
    rprpair Character string indicating the location and scale parameters to use. Default to
                "hl-disp", but may also be "med-mad". See Bilgic (2012).

Value

    siga2  Variance from cluster
    sigw2  Variance from subcluster
    sigmain2 Remaining variance not accounted for by variance of cluster and subcluster

Author(s)

    J. W. McKean and Y. K. Bilgic

References

    Y. K. Bilgic. Rank-based estimation and prediction for mixed effects models in nested designs.
See Also
rprmeddis, dispvar

Examples

# Load school data
data(schools)

# Fit fixed effects model with lmr
lmr.fit = lmr(y ~ age + sex, data=schools)

# Three level design
# Partition residuals into school and region effects with rpp algorithm
rpr(y ~ age + sex + (1 | school) + (1 | school:region), lmr.fit$ehat, schools)

# Two level design
# Estimate variance in residuals from school
rpr(y ~ age + sex + (1 | school), lmr.fit$ehat, schools)

rprmeddis

Rprmeddis

Description

Robust rank-based prediction algorithm that gets predictions for random errors in three-level nested design. It needs one location and scale estimators. Hodges-Lehmann location estimate and dispersion functional estimate pair is called with rprpair="hl-disp" -by default- ; median and MAD pair is called with rprpair="med-mad" in rlme().

Usage

rprmeddis(I, sec, mat, ehat, location, scale, rprpair = "hl-disp")

Arguments

I
sec
mat
ehat
location
scale
rprpair

Number of clusters.
A vector of subcluster numbers in clusters.
A matrix of numbers of observations in subclusters. Dimension is Ixmax(number ofsubclusters). Each row indicates one cluster.
The residuals that inherits random effects and error effect to be predicted.
If location = scale = 1 then use Median and MAD in RPP If location = scale = 2 then use HL & Dispvar in RPP Note: this is deprecated. You should specify the location & scale parameters by using the rprpair parameter.
1 means mad, 2 means disp as scale estimators
Character string indicating the location and scale parameters to use. Default to "hl-disp", but may also be "med-mad". See Bilgic (2012).
Details

The rprmeddisp() function yields predictions of random effects and errors vectors along with scale estimates in each level. This function was designed for three-level nested design. See rprmeddisp2() in the package, this is for two-level nested design.

Author(s)

Yusuf Bilgic <yekabe@hotmail.com>

References


See Also

rpr dispvar

schools

PISA Literacy Data

Description

The data in Program for International Assessment (PISA) on academic proficiency in schools around the world.

Format

A data frame with 334 observations on the following 6 variables.

y  a numeric vector indicating student literacy
socio  a numeric vector
sex  a numeric vector
age  a numeric vector
region  a numeric vector indicating four regions
school  a numeric vector indicating the schools within region

References

Examples

# The example takes a few seconds to run, so in order to
# save CRAN's testing time it has been commented out.
# To run, simply uncomment and execute.
#
# data(schools)
# rlme.fit = rlme(y ~ 1 + sex + age + (1 | region) + (1 | region:school),
# schools, method="gr")
# summary(rlme.fit)

### stanresidgr

#### Calculate Standard Residuals

Description

Standardizes the residuals obtained from the GR fitting.

Usage

stanresidgr(x, y, resid, delta = 0.8, param = 2, conf = 0.95)

Arguments

- **x**: Design matrix.
- **y**: Response vector.
- **resid**: Residuals obtained from the rank-based fitting.
- **delta**: See HM (2012).
- **param**: See HM (2012).
- **conf**: See HM (2012).

Author(s)

J. W. McKean

References


**summary.rlme**  

*rlme Summary*

**Description**

Summarizes a model fit from the rmle function

**Usage**

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

**Arguments**

- `object`: A list of class rlme
- `...`: not used

**Author(s)**

Herb Susmann <hps1@geneseo.edu>

**See Also**

- `rlme`
- `plot.rlme`

---

**wilonestep**  

*Wilcoxon estimate for independent linear models*

**Description**

This function gets weighted rank based fittings.

**Usage**

```r
wilonestep(y, x)
```

**Arguments**

- `y`: Response vector of nx1.
- `x`: Design matrix, pxn, without intercept.

**References**

wilstep

Wilcoxon One Step Rank-based Estimate in GR Method

Description

Gets weighted rank based fittings for nested designs.

Usage

wilstep(I, sec, mat, init = F, y, x, sigmaa2 = 1, sigmaw2 = 1,
       sigmae2 = 1, thetaold = c(0), eps = 1e-04, iflag2 = 0,
       rprpair = "hl-disp")

Arguments

I  
Number of clusters.
sec  
A vector of subcluster numbers in clusters.
mat  
A matrix of numbers of observations in subclusters. Dimension is Ixmax(number of subclusters). Each row indicates one cluster.
init  
boolean
y  
Response vector of nx1.
x  
Design matrix, pxn, without intercept.
sigmaa2  
Initial sigma for cluster in three-level design.
sigmaw2  
Initial sigma for subcluster in three-level design.
sigmae2  
Initial sigma for error in three-level design.
thetaold  
Initial input.
eps  
Epsilon value
iflag2  
y or n
rprpair  
Either ‘hl-disp’ or ‘med-mad’

Details

Initial inputs are from the independent model.

Author(s)

J. W. McKean and Y. K. Bilgic

References

Index

* datasets
  instruction, 11
  schools, 22
* models
  compare.fits, 4
  GEER_est, 7
  GR_est, 9
  LM_est, 14
  rlme, 18
  rlme-package, 2
* package
  rlme-package, 2
beta_var, 3
compare.fits, 4, 6
dispvar, 5, 22
fitdvcov, 4, 6
GEER_est, 7
getgrstplot, 8
getlmestplot, 9
GR_est, 9
hhbrwts_gr, 10
instruction, 11
JR_est, 12
LM_est, 14
lmr, 13
minimize_dispersion, 15
plot.rlme, 16, 24
rhosch, 17
rhosect, 17
rlme, 3, 13, 18, 24
rlme-package, 2
rpr, 20, 22
rprmeddis, 21
schools, 22
stanresidgr, 23
summary.rlme, 24
wilonestep, 24
wilstep, 25

26