Package ‘micemd’

July 9, 2017

Type Package

Title Multiple Imputation by Chained Equations with Multilevel Data

Version 1.1.0

Date 2017-07-06

Description Addons for the 'mice' package to perform multiple imputation using chained equations with two-level data. Includes imputation methods specifically handling sporadically and systemati- cally missing values. Imputation of continuous, binary or count variables are available. Follow- ing the recommendations of Audigier, V. et al (2017), the choice of the imputa- tion method for each variable can be facilitated by a default choice tuned according to the struc- ture of the incomplete dataset. Allows parallel calculation for 'mice'.

License GPL-2 | GPL-3

Depends R (>= 3.2.5), mice (>= 2.25)

Imports Matrix, graphics, utils, stats, MASS, parallel, nlme, lme4,
mvmeta (>= 0.4.7), jomo (>= 2.4-1), mvtnorm, digest, abind

Suggests VIM

RoxygenNote 6.0.0

NeedsCompilation no

Author Vincent Audigier [aut, cre] (INSERM ECSTRA team),
Matthieu Resche-Rigon [aut] (INSERM ECSTRA team)

Maintainer Vincent Audigier <vincent.audigier@univ-paris-diderot.fr>

Repository CRAN

Date/Publication 2017-07-09 16:04:51 UTC

R topics documented:

  micemd-package .......................................................... 2
  CHEM97Na .............................................................. 4
  find.defaultMethod ..................................................... 5
  IPDNa ................................................................. 7
  mice.impute.2l.2stage.bin ............................................ 9
Addons for the mice package to perform multiple imputation using chained equations with two-level data. Includes imputation methods specifically handling sporadically and systematically missing values (Resche-Rigon et al. 2013). Imputation of continuous, binary or count variables are available. Following the recommendations of Audigier, V. et al (2017), the choice of the imputation method for each variable can be facilitated by a default choice tuned according to the structure of the incomplete dataset. Allows parallel calculation for mice.

Details

Package: micemd
Type: Package
Version: 1.0.2
Date: 2017-06-29
License: GPL
LazyLoad: yes

Author(s)

Vincent Audigier, Matthieu Resche-Rigon
Maintainer: Vincent Audigier <vincent.audigier@univ-paris-diderot.fr>

References


See Also
mice

Examples

```r
require(lme4)
data(CHEM97Na)

ind.clust<-1 #index for the cluster variable

#initialisation of the argument predictorMatrix
temp<-mice(CHEM97Na, m=1, maxit=0)
temp.pred[ind.clust,ind.clust]<-0; temp.pred[-ind.clust,ind.clust]<- -2
temp.pred[temp.pred==1]<-2
predictor.matrix<-temp.pred

#initialisation of the argument method
method<-find.defaultMethod(CHEM97Na, ind.clust)

#multiple imputation by chained equations (parallel calculation) [time consuming]
#res.mice<-mice.par(CHEM97Na,predictorMatrix = predictor.matrix, method=method)

#check convergence
#plot(res.mice)

#analysis (apply a generalized linear mixed effects model to each imputed dataset)
#ana<-with(res.mice,expr=glmer(Score~Sex+GSCE+(1|School),
# family="poisson",
# control=glmerControl(optimizer = "bobyqa")))

#check the number of generated tables
```
An incomplete two-level dataset which consists of A/AS-level examination data from England

Description

This dataset is an extract of the CHEM97 dataset (Fielding, A. et al, 2003) dealing with point scores of 31,022 pupils grouped in 2,280 schools. CHEM97Na reports point score for Schools with more than 70 pupils only, i.e. 1681 pupils grouped in 18 schools. Systematically missing values and sporadically missing values have been added according to a missing completely at random (MCAR) mechanism (Little R.J.A. and Rubin D.B., 2002). Systematically missing values are values that are missing for all pupils of a same school, while sporadically missing values are values which are missing for an individual only (Resche-Rigon, et al 2013).

Usage

data("CHEM97Na")

Format

A data frame with 1681 observations on the following 5 variables.

School  a numeric indexing the School
Sex    a factor with levels M F
Age    a numeric indicating the age in months
GSCE   a numeric vector indicating the point score at the General Certificate of Secondary Education
Score  a numeric vector indicating the point score on A-level Chemistry in 1997

Details

For more details, see Fielding, A. et al (2003).

Source

Available at http://www.bristol.ac.uk/cmm/learning/mmsoftware/data-rev.html#chem97
find.defaultMethod

References

See Also
matrixplot

Examples

data(CHEM97Na)

#summary
summary(CHEM97Na)

#summary per School
by(CHEM97Na,CHEM97Na$School,summary)

find.defaultMethod Suggestion of conditional imputation models to use accordingly to the incomplete dataset

Description
Provides conditionnal imputation models to use for each column of the incomplete dataset according to the number of clusters, the number of individuals per cluster and the class of the variables.

Usage
find.defaultMethod(don.na, ind.clust, I.small = 7, ni.small = 100, prop.small = 0.4)

Arguments
don.na An incomplete data frame.
ind.clust A scalar indexes the variable corresponding to the cluster indicator.
I.small A scalar that is used as threshold to consider the number of observed clusters (fully observed or partially observed) as small. Default is I.small=7.
ni.small A scalar that is used as threshold to consider the number individuals per clusters (with observed values) as small. Default is ni.small=100.
prop.small A scalar that is used as threshold to consider the number of small clusters as small. Default is prop.small=0.4.
Details

Provides conditionnal imputation models to use for each column of the incomplete dataset according to the number of clusters, the number of individuals per cluster and the class of the variable (Audigier, V. et al 2017). Returned methods can be: 2l.stage.bin (binary), 2l.stage.norm (continuous), 2l.stage.pois (integer), 2l(glm.bin (binary), 2l(glm.norm (continuous), 2l(glm.pois (integer), 2l.jomo (continuous or binary). For a given variable, the method retained is chosen according to the following decision tree:

<table>
<thead>
<tr>
<th>Few observed clusters</th>
<th>Many observed clusters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Few observed values per cluster</td>
<td>2l(glm.norm)</td>
</tr>
<tr>
<td>continuous</td>
<td>2l.stage.norm</td>
</tr>
<tr>
<td>binary</td>
<td>2l.stage.bin</td>
</tr>
<tr>
<td>integer</td>
<td>2l.stage.pois</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Many observed clusters</th>
<th>Few observed values per cluster</th>
<th>Many observed values per cluster</th>
</tr>
</thead>
<tbody>
<tr>
<td>continuous</td>
<td>2l(glm.norm)</td>
<td>2l.stage.norm</td>
</tr>
<tr>
<td>binary</td>
<td>2l.jomo</td>
<td>2l.stage.bin</td>
</tr>
<tr>
<td>integer</td>
<td>2l(glm.pois)</td>
<td>2l.stage.pois</td>
</tr>
</tbody>
</table>

For instance, with few observed clusters (i.e. less than \( i_{small} \)), and many observed values per cluster (i.e. less than \( prop_{small} \) clusters with less than \( n_{i,small} \) observed values), imputation of a continuous variable according to the method 2l.stage.norm will be suggested.

Value

A vector of strings with length `ncol(data)`.

Author(s)

Vincent Audigier <vincent.audigier@univ-paris-diderot.fr>

References


See Also

mice, mice.par

Examples

data(CHEM97Na)

ind.clust<-1 # index for the cluster variable

# initialisation of the argument predictorMatrix
temp<-mice(CHEM97Na, m=1, maxit=0)
temp$pred[ind.clust, ind.clust] <- 8; temp$pred[-ind.clust, ind.clust] <- -2
temp$pred[temp$pred==1] <- 2
predictor.matrix<- temp$pred

# initialisation of the argument method
method<-find.defaultMethod(CHEM97Na, ind.clust)
print(method)

# multiple imputation by chained equations (parallel calculation)
# res.mice<-mice.par(CHEM97Na, m=3, predictorMatrix = predictor.matrix, method=method)

---

**IPDNa**

*An example of Individual Patient Data (IPD) meta-analysis with missing values.*

**Description**

This dataset is a simulated version of an IPD meta-analysis consisting of 28 studies focusing on risk factors in acute heart failure (GREAT, 2013). Each study includes a list of patient characteristics and potential risk factors. Each of them is incomplete, leading to sporadically missing values (Resche-Rigon, et al 2013). In addition, some variables have been collected on some studies only, leading to systematically missing values. More details on the original dataset are provided in Audigier et al. (2017). To mimic the real data, a general location model has been fitted on each study (Schafer, 1997). Then, each study has been generated according to the estimated parameters. Finally, missing values have been allocated similarly to the original dataset.
Usage

data("IPDNa")

Format

A data frame with 11685 observations on the following 10 variables.

centre a numeric indexing the center where the study is conducted

gender a factor with levels 0 1

bmi a numeric vector indicating the body mass index

age a numeric vector indicating the age

sbp a numeric vector indicating the systolic blood pressure

dbp a numeric vector indicating the diastolic blood pressure

hr a numeric vector indicating the heart rate

lvef a numeric vector indicating the ventricular ejection fraction

bnp a numeric vector indicating the level of the brain natriuretic peptide biomarker

afib a factor with levels 0 1 indicating the atrial fibrillation

Details

For more details, see Audigier et al. (2017)

Source


References


Examples

data(IPDNa)

#summary
summary(IPDNa)

#summary per study
by(IPDNa,IPDNa$centre,summary)
Imputation by a two-level logistic model based on a two-stage estimator

Description

Imputes univariate two-level binary variable from a logistic model. The imputation method is based on a two-stage estimator: at step 1, a logistic regression model is fitted to each observed cluster; at step 2, estimates obtained from each cluster are combined according to a linear random effect model.

Usage

mice.impute.2l.2stage.bin(y, ry, x, type, method_est = "mm", ...)

Arguments

- **y**: Incomplete data vector of length n
- **ry**: Vector of missing data pattern (FALSE=missing, TRUE=observed)
- **x**: Matrix (n x p) of complete covariates.
- **type**: Vector of length ncol(x) identifying random and class variables. Random variables are identified by a ’2’. The class variable (only one is allowed) is coded as ’-2’. Random variables also include the fixed effect.
- **method_est**: Vector of string given the version of the estimator to used. Choose method_est="reml" for restricted maximum likelihood estimator or method_est="mm" for the method of moments. By default method_est="mm".
- **...**: Other named arguments.

Details

Imputes univariate two-level continuous variable from a heteroscedastic normal model. The imputation method is based on a two-stage estimator: at step 1, a linear regression model is fitted to each observed cluster; at step 2, estimates obtained from each cluster are combined according to a linear random effect model. Two possibilities are available to combine estimates at stage 2: by default, parameters of the linear random effect model are estimated according to the method of moments (MM), otherwise, parameters of the linear random effect model can be estimated according to the restricted maximum likelihood estimator (REML). The variability on the parameters of the imputation is propagated according to an asymptotic strategy requiring notably a large number of clusters. Compared to the REML version, the MM version is quicker to perform, but it provides less theoretical garanties. Nevertheless, simulations studies show that both versions lead to similar inferences (Audigier et al, 2017; Resche-Rigon, M. and White, I. R., 2016).

Value

A vector of length nmis with imputations.
Author(s)

Vincent Audigier <vincent.audigier@univ-paris-diderot.fr>

References


See Also

mice, mice.impute.2l.glm.bin, mice.impute.2l.jomo

mice.impute.2l.2stage.norm

Imputation by a two-level heteroscedastic normal model based on a two-stage estimator

Description

Imputes univariate two-level continuous variable from a heteroscedastic normal model. The imputation method is based on a two-stage estimator: at step 1, a linear regression model is fitted to each observed cluster; at step 2, estimates obtained from each cluster are combined according to a linear random effect model.

Usage

mice.impute.2l.2stage.norm(y, ry, x, type, method_est = "mm", ...)

Arguments

y Incomplete data vector of length n
ry Vector of missing data pattern (FALSE=missing, TRUE=observed)
x Matrix (n x p) of complete covariates.
type Vector of length ncol(x) identifying random and class variables. Random variables are identified by a '2'. The class variable (only one is allowed) is coded as '-2'. Random variables also include the fixed effect.
method_est Vector of string given the version of the estimator to used. Choose method_est="reml" for restricted maximum likelihood estimator or method_est="mm" for the method of moments. By default method_est="mm".
... Other named arguments.
Details

Imputes univariate two-level continuous variable from a heteroscedastic normal model. The imputation method is based on a two-stage estimator: at step 1, a linear regression model is fitted to each observed cluster; at step 2, estimates obtained from each cluster are combined according to a linear random effect model. Two possibilities are available to combine estimates at stage 2: by default, parameters of the linear random effect model are estimated according to the method of moments (MM), otherwise, parameters of the linear random effect model can be estimated according to the restricted maximum likelihood estimator (REML). The variability on the parameters of the imputation is propagated according to an asymptotic strategy requiring notably a large number of clusters. Compared to the REML version, the MM version is quicker to perform, but it provides less theoretical guarantees. Nevertheless, simulations studies show that both versions lead to similar inferences (Resche-Rigon, M. and White, I. R. (2016)).

Value

A vector of length nmis with imputations.

Author(s)

Vincent Audigier <vincent.audigier@univ-paris-diderot.fr>

References


See Also

mice,mice.impute.2l.2stage.pmm,mice.impute.2l(glm).norm,mice.impute.2l.jomo
Usage

```r
class2stage.pmm(y, ry, x, type, 
    method_est = "mm", 
    incluster = FALSE, 
    k = 5, ...)
```

Arguments

- `y` Incomplete data vector of length `n`
- `ry` Vector of missing data pattern (FALSE=missing, TRUE=observed)
- `x` Matrix (n \times p) of complete covariates.
- `type` Vector of length `n\text{col}(x)` identifying random and class variables. Random variables are identified by a '2'. The class variable (only one is allowed) is coded as '-2'. Random variables also include the fixed effect.
- `method_est` Vector of string given the version of the estimator to used. Choose method_est="reml" for restricted maximum likelihood estimator or method_est="mm" for the method of moments. By default method_est="mm".
- `incluster` Boolean indicating if the imputed values are drawn from the cluster or from the full dataset. By default imputed values are drawn from all available clusters incluster=FALSE.
- `k` The size of the donor pool among which a draw is made. The default is k = 5.
- `...` Other named arguments.

Details

Imputes univariate two-level continuous variable from observed values. The imputation method is based on a two-stage estimator: at step 1, a linear regression model is fitted to each observed cluster; at step 2, estimates obtained from each cluster are combined according to a linear random effect model. To combine estimates at stage 2, parameters of the linear random effect model are estimated according to the method of moments or according to the restricted maximum likelihood estimator. The variability on the parameters of the imputation is propagated according to an asymptotic strategy requiring notably a large number of clusters. The sample variability is reflected by using a predictive mean matching approach, meaning that missing values are imputed by a draw from observed values. The pool of k donors is defined according to the Manhattan distance between the prediction of the observation which is imputed and the predictions of other available observations (matching of type 2). The pool can be restricted to the cluster of the individual that is imputed or from all clusters. By drawing values inside the cluster, the heteroscedasticity assumption is preserved. Otherwise, the sample variability of imputed values is the same for all clusters, which strengthen the homoscedasticity assumption. Among the pool of k donors, the selected one is drawn at random.

Value

Numeric vector of length sum(!ry) with imputations
Note

This method is experimental.

Author(s)

Vincent Audigier <vincent.audigier@univ-paris-diderot.fr>

References


See Also

mice.impute.2l.2stage.norm

Imputation by a two-level Poisson model based on a two-stage estimator

Description

Imputes univariate two-level count variable from a Poisson model. The imputation method is based on a two-stage estimator: at step 1, a Poisson regression model is fitted to each observed cluster; at step 2, estimates obtained from each cluster are combined according to a linear random effect model.

Usage

mice.impute.2l.2stage.pois(y, ry, x, type, method_est = "mm", ...)

Arguments

y

Incomplete data vector of length n

ry

Vector of missing data pattern (FALSE=missing, TRUE=observed)

x

Matrix (n x p) of complete covariates.

type

Vector of length ncol(x) identifying random and class variables. Random variables are identified by a `2`. The class variable (only one is allowed) is coded as `-2`. Random variables also include the fixed effect.
method_est Vector of string given the version of the estimator to used. Choose method_est="reml" for restricted maximum likelihood estimator or method_est="mm" for the method of moments. By default method_est="mm".

... Other named arguments.

Details

Imputes univariate two-level count variable from a Poisson model. The imputation method is based on a two-stage estimator: at step 1, a Poisson regression model is fitted to each observed cluster; at step 2, estimates obtained from each cluster are combined according to a linear random effect model. Two possibilities are available to combine estimates at stage 2: by default, parameters of the linear random effect model are estimated according to the method of moments (MM), otherwise, parameters of the linear random effect model can be estimated according to the restricted maximum likelihood estimator (REML). The variability on the parameters of the imputation is propagated according to an asymptotic strategy requiring large samples and a large number of clusters. Compared to the REML version, the MM version is quicker to perform, but it provides less theoretical garanties. Nevertheless, simulations studies show that both versions lead to similar inferences (Audigier et al, 2017; Resche-Rigon and White, 2016).

Value

A vector of length nmis with imputations.

Author(s)

Vincent Audigier <vincent.audigier@univ-paris-diderot.fr>

References


See Also

mice,mice.impute.2l.glm.pois
Description

Imputes univariate missing data using a Bayesian logistic mixed model based on non-informative prior distributions. The method is dedicated to a binary outcome stratified in several clusters. Should be used with few clusters and few individuals per cluster. Can be very slow to perform otherwise.

Usage

mice.impute.2l.glm.bin(y, ry, x, type, ...)

Arguments

y
Incomplete data vector of length n

ry
Vector of missing data pattern (FALSE=missing, TRUE=observed)

x
Matrix (n x p) of complete covariates.

type
Vector of length ncol(x) identifying random and class variables. Random variables are identified by a '2'. The class variable (only one is allowed) is coded as '2'. Random variables also include the fixed effect.

...
Other named arguments.

Details

Imputes univariate missing data using a Bayesian logistic mixed model based on non-informative prior distributions. The variability on the parameters of the imputation is propagated according to an explicit Bayesian modelling. More precisely, improper prior distributions are used for regression coefficients and covariance matrix of random effects. The method is recommended for datasets with a small number of clusters and a small number of individuals per cluster. Otherwise, the method can be very slow to perform.

Value

A vector of length nmis with imputations.

Author(s)

Vincent Audigier <vincent.audigier@univ-paris-diderot.fr> from the R code of Shahab Jolani.

References


Imputation of univariate missing data using a Bayesian linear mixed model based on non-informative prior distributions

Description

Imputes univariate missing data using a Bayesian linear mixed model based on non-informative prior distributions. The method is dedicated to a continuous outcome stratified in several clusters. Should be used with few clusters and few individuals per cluster. Can be very slow to perform otherwise.

Usage

mice.impute.2l.glm.norm(y, ry, x, type, ...)

Arguments

- `y`: Incomplete data vector of length n
- `ry`: Vector of missing data pattern (FALSE=missing, TRUE=observed)
- `x`: Matrix (n x p) of complete covariates.
- `type`: Vector of length ncol(x) identifying random and class variables. Random variables are identified by a '2'. The class variable (only one is allowed) is coded as '-2'. Random variables also include the fixed effect.
- `...`: Other named arguments.

Details

Imputes univariate two-level continuous variable from a homoscedastic normal model. The variability on the parameters of the imputation is propagated according to an explicit Bayesian modelling. More precisely, improper prior distributions are used for regression coefficients and variances components. The method is recommended for datasets with a small number of clusters and a small number of individuals per cluster. Otherwise, confidence intervals after applying analysis method on the multiply imputed dataset tend to be anti-conservative. In addition, the imputation can be highly time consuming otherwise.

Value

A vector of length nmis with imputations.

Author(s)

Vincent Audigier <vincent.audigier@univ-paris-diderot.fr> from the R code of Shahab Jolani.
References


See Also

mice, mice.impute.2l.2stage.norm, mice.impute.2l.jomo

mice.impute.2l.glm.pois

Imputation of count variable using a Bayesian mixed model based on non-informative prior distributions

Description

Imputes univariate missing data using a Bayesian mixed model (Poisson regression) based on non-informative prior distributions. The method is dedicated to a count outcome stratified in several clusters. Should be used with few clusters and few individuals per cluster. Can be very slow to perform otherwise.

Usage

mice.impute.2l.glm.pois(y, ry, x, type,...)

Arguments

y
Incomplete data vector of length n
ry
Vector of missing data pattern (FALSE=missing, TRUE=observed)
x
Matrix (n x p) of complete covariates.
type
Vector of length ncol(x) identifying random and class variables. Random variables are identified by a '2'. The class variable (only one is allowed) is coded as '-2'. Random variables also include the fixed effect.
... Other named arguments.

Details

Imputes univariate missing data using a Bayesian mixed model (Poisson regression) based on non-informative prior distributions. The variability on the parameters of the imputation is propagated according to an explicit Bayesian modelling. More precisely, improper prior distributions are used for regression coefficients and variances components. The method is recommended for datasets with a small number of clusters and a small number of individuals per cluster. Otherwise, the method can be very slow to perform.
Value

A vector of length nmis with imputations.

Author(s)

Vincent Audigier <vincent.audigier@univ-paris-diderot.fr> from the R code of Shahab Jolani.

References


See Also

mice, mice.impute.2l.2stage.pois

mice.impute.2l.jomo

Imputation of univariate missing data by a Bayesian multivariate generalized model based on conjugate priors

Description

Univariate imputation by a Bayesian multivariate generalized model based on conjugate priors. Can be used for a continuous or binary incomplete variable. For continuous variables, the modelling assumes heteroscedasticity for errors. For a binary variable, a probit link and a latent variables framework are used. The method should be used for a variable with large number of clusters and a large number of individuals per cluster.

Usage

mice.impute.2l.jomo(y, ry, x, type, nburn = 200, ...)

Arguments

y  Incomplete data vector of length n
ry  Vector of missing data pattern (FALSE=missing, TRUE=observed)
x  Matrix (n x p) of complete covariates.
type  Vector of length ncol(x) identifying random and class variables. Random variables are identified by a ’2’. The class variable (only one is allowed) is coded as ’-2’. Random variables also include the fixed effect.
nburn  A scalar indicating the number of iterations for the Gibbs sampler. Default is nburn=200
...  Other named arguments.
Details

Contrary to the approach developped in the R jomo package, the imputation is here sequentially performed through a FCS approach, instead of imputing all variables simulatenously. The motivation for such a method is that jomo presents some advantages over other imputation methods, but not always for any type of variables (binary or continuous). By proposing a FCS version of jomo, we allow imputation of mixed variables (continuous and binary), while taking the best of jomo and of other imputation methods. To impute one variable according to this method, other variables are assumed to be full, like in any FCS approach. The imputation function is a direct use of the R function jomo1ran from the jomo package. The argument meth is tuned to "random" to allow covariance matrices drawn from an inverse Wishart distribution. Only intercept are considered in covariates (X=\text{NULL} and Z=\text{NULL}), while the multivariate outcome corresponds to all variables of the datasets.

Value

A vector of length nmis with imputations.

Author(s)

Vincent Audigier <vincent.audigier@univ-paris-diderot.fr> from the R code of Matteo Quartagno.

References


See Also

mice,jomo1ran

Description

Parallel calculations for Multivariate Imputation by Chained Equations using the R package parallel.
Usage

mice.par(don.na, m = 5, method = vector("character", length = ncol(don.na)),
predictorMatrix = (1 - diag(1, ncol(don.na))[,1]),
visitSequence = (1:ncol(don.na))[apply(is.na(don.na), 2, any)],
form = vector("character", length = ncol(don.na)),
post = vector("character", length = ncol(don.na)),
defaultMethod = c("pmm", "logreg", "polyreg", "polr"),
maxit = 5, diagnostics = TRUE, seed = NA, imputationMethod = NULL,
defaultImputationMethod = NULL, data.init = NULL,
nnodes = 5, path.outfile = NULL, ...)

Arguments

don.na
A data frame or a matrix containing the incomplete data. Missing values are
coded as NA.
m
Number of multiple imputations. The default is m=5.
method
Can be either a single string, or a vector of strings with length ncol(data),
specifying the elementary imputation method to be used for each column in
data. If specified as a single string, the same method will be used for all columns.
The default imputation method (when no argument is specified) depends on the
measurement level of the target column and are specified by the defaultMethod
argument. Columns that need not be imputed have the empty method "". See
details for more information.
predictorMatrix
A square matrix of size ncol(data) containing 0/1 data specifying the set of
predictors to be used for each target column. Rows correspond to target variables
(i.e. variables to be imputed), in the sequence as they appear in data. A value of
'1' means that the column variable is used as a predictor for the target variable
(in the rows). The diagonal of predictorMatrix must be zero. The default for
predictorMatrix is that all other columns are used as predictors (sometimes
called massive imputation). Note: For two-level imputation codes '2' and '-2'
are also allowed.
visitSequence
A vector of integers of arbitrary length, specifying the column indices of the
visiting sequence. The visiting sequence is the column order that is used to
impute the data during one pass through the data. A column may be visited more
than once. All incomplete columns that are used as predictors should be visited,
or else the function will stop with an error. The default sequence 1:ncol(data)
implies that columns are imputed from left to right. It is possible to specify one
of the keywords 'roman' (left to right), 'arabic' (right to left), 'monotone'
(sorted in increasing amount of missingness) and 'revmonotone' (reverse of
monotone). The keyword should be supplied as a string and may be abbreviated.
form
A vector of strings with length ncol(data), specifying formulae. Each string
is parsed and executed within the sampler() function to create terms for the
predictor. The default is to do nothing, indicated by a vector of empty strings
"". The main value lies in the easy specification of interaction terms. The user
must ensure that the set of variables in the formula match those in predictors.
A vector of strings with length `ncol(data)`, specifying expressions. Each string is parsed and executed within the `sampler()` function to postprocess imputed values. The default is to do nothing, indicated by a vector of empty strings `''`.

defaultMethod

A vector of three strings containing the default imputation methods for numerical columns, factor columns with 2 levels, and columns with (unordered or ordered) factors with more than two levels, respectively. If nothing is specified, the following defaults will be used: `pmm`, predictive mean matching (numeric data) `logreg`, logistic regression imputation (binary data, factor with 2 levels) `polyreg`, polytomous regression imputation for unordered categorical data (factor >= 2 levels) `polr`, proportional odds model for (ordered, >= 2 levels)

maxit

A scalar giving the number of iterations. The default is 5.

diagnostics

A Boolean flag. If `TRUE`, diagnostic information will be appended to the value of the function. If `FALSE`, only the imputed data are saved. The default is `TRUE`.

seed

An integer that is used as argument by the `set.seed()` for offsetting the random number generator. Default is to leave the random number generator alone.

imputationMethod

Same as `method` argument. Included for backwards compatibility.

defaultImputationMethod

Same as `defaultMethod` argument. Included for backwards compatibility.

data.init

A data frame of the same size and type as `data`, without missing data, used to initialize imputations before the start of the iterative process. The default `NULL` implies that starting imputation are created by a simple random draw from the data. Note that specification of `data.init` will start the `m` Gibbs sampling streams from the same imputations.

nnodes

A scalar indicating the number of nodes for parallel calculation. Default value is 5.

path.outfile

A vector of strings indicating the path for redirection of print messages. Default value is `NULL`, meaning that silent imputation is performed. Otherwise, print messages are saved in the files `path.outfile/output.txt`. One file per node is generated.

Details

Performs multiple imputation of `m` tables in parallel by generating `m` seeds, and then by performing multiple imputation by chained equations in parallel from each one. The output is the same as the `mice` function of the mice package.

Value

Returns an S3 object of class `mids` (multiply imputed data set)

Author(s)

Vincent Audigier <vincent.audigier@univ-paris-diderot.fr>
References

<doi:10.18637/jss.v045.i03>


See Also

*mice*, *parallel*

Examples

```
#CHEM97Na (1681 observations and 5 variables)

data(CHEM97Na)

ind.clust<-1 #index for the cluster variable

#initialisation of the argument predictorMatrix
temp<-mice(CHEM97Na,m=1,maxit=0)
temp$pred[ind.clust,ind.clust]<-0
temp$pred[-ind.clust,ind.clust]<-2
temp$pred[temp$pred==1]<-2
predictor.matrix<-temp$pred

#initialisation of the argument method
method<-find.defaultMethod(CHEM97Na,ind.clust)

#multiple imputation by chained equations (parallel calculation) [1 minute]
#(the imputation process can be followed by open the files output.txt in the working directory)
res.mice<-mice.par(CHEM97Na,
# predictorMatrix = predictor.matrix,
# method=method,
# path.outfile=getwd())
```
#multiple imputation by chained equations (without parallel calculation) [4.8 minutes]
#res.mice<-mice(CHEM97Na,
  #    predictorMatrix = predictor.matrix,
  #    method=method)

#IPDNa (11685 observations and 10 variables)

data(IPDNa)

ind.clust<-1#index for the cluster variable

#initialisation of the argument predictorMatrix
temp<-mice(IPDNa,m=1,maxit=0)
temp$pred[ind.clust,ind.clust]<0
temp$pred[-ind.clust,ind.clust]< -2
temp$pred[temp$pred==1]<-2
predictor.matrix<-temp$pred

#initialisation of the argument method
method<-find.defaultMethod(IPDNa,ind.clust)

#multiple imputation by chained equations (parallel calculation)

#res.mice<-mice.par(IPDNa,
  #    predictorMatrix = predictor.matrix,
  #    method=method,
  #    path.outfile=getwd())

plot.mira

Graphical investigation for the number of generated datasets

Description

The plot method for a mira object plots the confidence interval length against the number of multiply imputed datasets from 2 to m. This is a graphical tool to check if the variability due to the simulation of the multiple imputation process can be substantially reduced by increasing the number of generated datasets m.

Usage

## S3 method for class 'mira'
plot(x, ...)
plot.mira

Arguments

x  An object of class mira.
... Extra arguments for plot.mira

Author(s)

Vincent Audigier <vincent.audigier@univ-paris-diderot.fr>

References


See Also

mice, mira

Examples

require(nlme)
data(CHEM97Na)

ind.clust<-1 # index for the cluster variable

#initialisation of the argument predictorMatrix
temp<-mice(CHEM97Na,m=1,maxit=0)
temp$pred[ind.clust,ind.clust]<=0;temp$pred[-ind.clust,ind.clust]<= -2
temp$pred[temp$pred==1]<=2
predictor.matrix<-temp$pred

#initialisation of the argument method
method<-("","21.2stage.bin","21.2stage.pois","21.2stage.norm","") # quickest methods

#multiple imputation by chained equations (parallel calculation)
#res.mice<-mice.par(CHEM97Na,m=15,predictorMatrix = predictor.matrix,method=method)

#analysis (apply a linear mixed effects model to each imputed dataset)
#ana<-with(res.mice,expr=1me(fixed=formula(Score~Sex+GSCE+Age),
# random=formula(~1|School),method="REML",
# control=list(maxIter=100,msMaxIter=100,niterEM=25)))

# graphical investigation for the number of generated datasets m
# plot(ana)
Index

*Topic datasets
  CHEM97Na, 4
  IPDNa, 7

*Topic imputation
  micemd-package, 2

*Topic meta-analyse
  micemd-package, 2

*Topic mice
  find.defaultMethod, 5
  mice.impute.21.2stage.bin, 9
  mice.impute.21.2stage.norm, 10
  mice.impute.21.2stage.pois, 13
  mice.impute.21.glm.bin, 14
  mice.impute.21.glm.norm, 16
  mice.impute.21.glm.pois, 17
  mice.impute.21.jomo, 18
  mice.par, 19

*Topic multilevel
  micemd-package, 2

*Topic package
  micemd-package, 2

CHEM97Na, 4

find.defaultMethod, 5

IPDNa, 7

jomo1ran, 19

matrixplot, 5
mice, 3, 7, 10, 11, 14, 16–19, 22, 24
mice.impute.21.2stage.bin, 9, 16
mice.impute.21.2stage.norm, 10, 13, 17
mice.impute.21.2stage.pmm, 11, 11
mice.impute.21.2stage.pois, 13, 18
mice.impute.21.glm.bin, 10, 14
mice.impute.21.glm.norm, 11, 16
mice.impute.21.glm.pois, 14, 17
mice.impute.21.jomo, 10, 11, 16, 17, 18
mice.par, 7, 19

micsdm-package, 2
mids, 21
mira, 24
parallel, 22
plot.mira, 23, 24