

Package ‘miceMNAR’

May 29, 2018

Type Package

Title Missing not at Random Imputation Models for Multiple Imputation
by Chained Equation

Version 1.0.1

Date 2018-05-28

Description Provides imputation models and functions for binary or continuous Missing Not At Random (MNAR) outcomes through the use of the 'mice' package. The `mice.impute.hecknorm()` function provides imputation model for continuous outcome based on Heckman's model also named sample selection model as described in Galimard et al (2016) <doi:10.1002/sim.6902>. The `mice.impute.heckprob()` function provides imputation model for binary outcome based on bivariate probit model.

License GPL-2 | GPL-3

Depends R (>= 3.2.1), mice (>= 3.0.0)

Imports stats, mvtnorm, pbivnorm, GJRM, sampleSelection

RoxygenNote 6.0.1

NeedsCompilation no

Author Jacques-Emmanuel Galimard [aut, cre] (INSERM, U1153, ECSTRA team),
Matthieu Resche-Rigon [aut] (INSERM, U1153, ECSTRA team)

Maintainer Jacques-Emmanuel Galimard <jacques-emmanuel.galimard@inserm.fr>

Repository CRAN

Date/Publication 2018-05-29 07:09:25 UTC

R topics documented:

<code>miceMNAR-package</code>	2
<code>generate_JointModelEq</code>	3
<code>mice.impute.hecknorm</code>	4
<code>mice.impute.hecknorm2step</code>	6
<code>mice.impute.heckprob</code>	8
<code>MNARargument</code>	10

Index	15
--------------	-----------

miceMNAR-package	<i>Missing not at Random Imputation Models for Multiple Imputation by Chained Equation</i>
------------------	--

Description

Provides imputation models and functions for binary or continuous Missing Not At Random (MNAR) outcomes through the use of the 'mice' package. The `mice.impute.hecknorm()` function provides imputation model for continuous outcome based on Heckman's model also named sample selection model as described in Galimard et al (2016) <doi:10.1002/sim.6902>. The `mice.impute.heckprob()` function provides imputation model for binary outcome based on bivariate probit model.

As these two previous imputation models require to specify a selection and an outcome equation, `mice()` function has to be adapted using `MNARargument()`.

Details

Index of help topics:

MNARargument	Function providing modified arguments for imputation of Missing Not At Random (MNAR) outcomes using 'mice()' function of the "mice" package
generate_JointModelEq	Generation of an empty matrix for sample selection model
mice.impute.hecknorm	Imputation by Heckman's model for continuous outcome with Missing Not At Random mechanism using one-step estimator
mice.impute.hecknorm2step	Imputation by Heckman's model for continuous Missing Not At Random outcome using a two-step estimator
mice.impute.heckprob	Imputation by bivariate probit sample selection model for binary Missing Not At Random outcome
miceMNAR-package	Missing not at Random Imputation Models for Multiple Imputation by Chained Equation

Warning

This package is only validated for the imputation of MNAR outcome. However, it is implemented to impute several MNAR variables in the same process. Such implementation must be realised carefully.

Author(s)

Jacques-Emmanuel Galimard [aut, cre] (INSERM, U1153, ECSTRA team), Matthieu Resche-Rigon [aut] (INSERM, U1153, ECSTRA team)

Maintainer: Jacques-Emmanuel Galimard <jacques-emmanuel.galimard@inserm.fr>

References

Galimard, J.-E., Chevret, S., Protopopescu, C., and Resche-Rigon, M. (2016) A multiple imputation approach for MNAR mechanisms compatible with Heckman's model. *Statist. Med.*, 35: 2907-2920. doi:10.1002/sim.6902.

See Also

[mice](#) [copulaSampleSel](#) [SemiParBIV](#)
[selection](#)

generate_JointModelEq *Generation of an empty matrix for sample selection model*

Description

This function generates a matrix of 0 with rows corresponding to the names of variables included in the dataset and column corresponding to selection and outcome equations for each MNAR outcome. The user has to fill this matrix with 1 for variable included in selection and/or outcome equations of MNAR outcome imputation models.

Usage

```
generate_JointModelEq(varMNAR, data)
```

Arguments

varMNAR	The name of MNAR outcome to be imputed.
data	The dataset used for classical multiple imputation by chained equation and additional variables necessary for MNAR imputation models.

Details

Be careful to not define the same selection and outcome equations for MNAR imputation models. A constraint of the sample selection model implies the inclusion of different sets of covariates, which may or not be nested in the selection equation and the outcome equation, to avoid collinearity issues. It has been recommended to include at least a supplementary variable in the selection equation. This variable should be known to be unlinked directly to the outcome.

Value

A matrix such as:

- Rows correspond to names of variables included in the data argument
- Columns correspond to selection and outcome equations for each MNAR outcome imputation models

Each column is named as "MNAR outcome" names adding "_var_sel" or "_var_out" corresponding to variables included in selection and outcome equation respectively.

Author(s)

Jacques-Emmanuel Galimard

See Also

[miceMNAR hiv](#)

Examples

```
# Import dataset with a suspected MNAR mechanism
require(GJRM)
data("hiv")

# We select only one region (lusuka) and 5 variables
lusuka <- hiv[hiv$region==5,c("hiv", "age", "marital", "condom", "smoke")]

# Categorical variables have to be recoded as factor
lusuka$hiv <- as.factor(lusuka$hiv)

# Specify a selection (missing data mechanism) and an outcome equation (analyse model)

# Generate an empty matrix
JointModelEq <- generate_JointModelEq(data=lusuka,varMNAR = "hiv")

# Fill in with 1 for variable included in equations
JointModelEq[, "hiv_var_sel"] <- c(0,1,1,1,1)
# This indicates that age, marital, condom and smoke are included in the selection equation of hiv
JointModelEq[, "hiv_var_out"] <- c(0,1,1,1,0)
# This indicates that age, marital and condom are included in the outcome equation of hiv
```

mice.impute.hecknorm *Imputation by Heckman's model for continuous outcome with Missing Not At Random mechanism using one-step estimator*

Description

Imputes univariate missing continuous outcome using Heckman's model with a one-step maximum likelihood estimator.

Usage

```
mice.impute.hecknorm(y, ry, x, JointModelEq, control, ...)
```

Arguments

y	Incomplete data vector of length n.
ry	Vector of missing data pattern of length n (FALSE=missing, TRUE=observed).
x	Matrix (n x p) of complete covariates.
JointModelEq	Output JointModelEq provided by MNAArgument() function.
control	Output control provided by MNAArgument() function.
...	Other named arguments.

Details

Imputation of continuous MNAR outcome using Heckman's model. This imputation model requires to define a selection and an outcome equation for the imputation model. The imputation algorithm consists of the following steps:

1. Fit the one-step Heckman's model and compute the posterior distribution of parameters
2. Draw imputation parameters from their posterior distribution
3. Draw imputed values from their predictive distribution

Value

A vector of length `nmiss` with imputations.

Author(s)

Jacques-Emmanuel Galimard and Matthieu Resche-Rigon

See Also

[miceMNAR](#)

Examples

```
require("GJRM")
require(mvtnorm)

# Generation of a simulated dataset with MNAR mechanism on a continuous outcome

X1 <- rnorm(500,0,1)
X2 <- rbinom(500,1,0.5)
X3 <- rnorm(500,1,0.5)

errors <- rmvnorm(500,mean=c(0,0),sigma=matrix(c(1,0.3,0.3,1),nrow=2,byrow=TRUE))

Y <- X1+X2+errors[,1]
Ry <- ifelse(0.66+1*X1-0.5*X2+X3+errors[,2]>0,1,0)

Y[Ry==0] <- NA
```

```

simul_data <- data.frame(Y,X1,X2,X3)

JointModelEq <- generate_JointModelEq(data=simul_data,varMNAR = "Y")

JointModelEq[, "Y_var_sel"] <- c(0,1,1,1)
JointModelEq[, "Y_var_out"] <- c(0,1,1,0)

arg <- MNARargument(data=simul_data,varMNAR="Y",JointModelEq=JointModelEq)

## Not run: imputation2 <- mice(data = arg$data_mod,
  method = arg$method,
  predictorMatrix = arg$predictorMatrix,
  JointModelEq=arg$JointModelEq,
  control=arg$control,
  maxit=1,m=5)

analysis2 <- with(imputation,lm(Y~X1+X2+X3))
result2 <- pool(analysis2)
summary(result2)
## End(Not run)

```

```
mice.impute.hecknorm2step
```

Imputation by Heckman's model for continuous Missing Not At Random outcome using a two-step estimator

Description

Imputes univariate continuous Missing Not At Random (MNAR) outcome using Heckman's model with a two-step estimator.

Usage

```
mice.impute.hecknorm2step(y, ry, x, JointModelEq, control, ...)
```

Arguments

y	Incomplete data vector of length n.
ry	Vector of missing data pattern of length n (FALSE=missing, TRUE=observed).
x	Matrix (n x p) of complete covariates.
JointModelEq	Output JointModelEq provided by MNARargument() function.
control	Output control provided by MNARargument() function.
...	Other named arguments.

Details

Imputation of continuous MNAR outcome using two-step Heckman's model (Galimard, 2016). This imputation model requires to define a selection and an outcome equation. The imputation algorithm consists of the following steps:

1. Compute Inverse Mill's Ratio (IMR) from the selection equation probit model
2. Include IMR as predictor in the imputation model
3. Draw imputation parameters using approximate proper imputation for the linear model and adding the Heckman variance correction as detailed in Galimard et al (2016)
4. Draw imputed values from their predictive distribution

Value

A vector of length `nmi.s` with imputations.

Note

`mice.impute.hecknorm()` should be preferred as `mice.impute.hecknorm2step()`. Accordingly, for continuous outcome, `mice.impute.hecknorm()` is a default. However, it is possible to modify the method argument by hand.

Author(s)

Jacques-Emmanuel Galimard and Matthieu Resche-Rigon

References

Galimard, J.-E., Chevret, S., Protopopescu, C., and Resche-Rigon, M. (2016) A multiple imputation approach for MNAR mechanisms compatible with Heckman's model. *Statist. Med.*, 35: 2907-2920. doi:10.1002/sim.6902.

See Also

[miceMNAR](#)

Examples

```
require("GJRM")
require(mvtnorm)
require(sampleSelection)

# Generation of a simulated dataset with MNAR mechanism on a continuous outcome

X1 <- rnorm(500,0,1)
X2 <- rbinom(500,1,0.5)
X3 <- rnorm(500,1,0.5)

errors <- rmvnorm(500,mean=c(0,0),sigma=matrix(c(1,0.3,0.3,1),nrow=2,byrow=TRUE))
```

```

Y <- X1+X2+errors[,1]
Ry <- ifelse(0.66+1*X1-0.5*X2+X3+errors[,2]>0,1,0)

Y[Ry==0] <- NA

simul_data <- data.frame(Y,X1,X2,X3)

JointModelEq <- generate_JointModelEq(data=simul_data,varMNAR = "Y")

JointModelEq[,"Y_var_sel"] <- c(0,1,1,1)
JointModelEq[,"Y_var_out"] <- c(0,1,1,0)

arg <- MNARargument(data=simul_data,varMNAR="Y",JointModelEq=JointModelEq)
arg$method["Y"] <- "hecknorm2step"

## Not run: imputation3 <- mice(data = arg$data_mod,
  method = arg$method,
  predictorMatrix = arg$predictorMatrix,
  JointModelEq=arg$JointModelEq,
  control=arg$control,
  maxit=1,m=5)

analysis3 <- with(imputation3,lm(Y~X1+X2+X3))
result3 <- pool(analysis3)
summary(result3)
## End(Not run)

```

mice.impute.heckprob *Imputation by bivariate probit sample selection model for binary Missing Not At Random outcome*

Description

Imputes univariate binary Missing Not At Random (MNAR) outcome using the bivariate probit sample selection model.

Usage

```
mice.impute.heckprob(y, ry, x, JointModelEq, control, ...)
```

Arguments

y	Incomplete data vector of length n.
ry	Vector of missing data pattern of length n (FALSE=missing, TRUE=observed).
x	Matrix (n x p) of complete covariates.
JointModelEq	Output JointModelEq provided by MNARargument() function.
control	Output control provided by MNARargument() function.
...	Other named arguments.

Details

Imputation of binary MNAR outcome using the bivariate probit sample selection model. This imputation model requires defining a selection and an outcome equation for the bivariate probit model. The imputation algorithm consists of the following steps:

1. Fit the bivariate probit model and compute the posterior distribution of parameters
2. Draw imputation parameters from their posterior distributions
3. Draw imputed values from their predictive distribution

Value

A vector of length `nmi`s with imputations (0 or 1).

Author(s)

Jacques-Emmanuel Galimard and Matthieu Resche-Rigon

See Also

[miceMNAR](#)

Examples

```
# Import dataset with a suspected MNAR mechanism
require("GJRM")
require(mvtnorm)
require(pbivnorm)
data("hiv")

# We select only one region (lusuka) and 5 variables
lusuka <- hiv[hiv$region==5,c("hiv", "age", "marital", "condom", "smoke")]

# Categorical variables have to be recoded as factor
lusuka$hiv <- as.factor(lusuka$hiv)

# Specify a selection (missing data mechanism) and an outcome equation (analyse model)

# Generate an empty matrix

JointModelEq <- generate_JointModelEq(data=lusuka,varMNAR = "hiv")

# Fill in with 1 for variable included in equations
JointModelEq[,"hiv_var_sel"] <- c(0,1,1,1,1)
JointModelEq[,"hiv_var_out"] <- c(0,1,1,1,0)

# Generation of argument for MNAR imputation model in "mice()" function
arg <- MNARargument(data=lusuka,varMNAR="hiv",JointModelEq=JointModelEq)

# Imputation using mice() function
# Values returned have to be included in the "mice()" function as argument:
```

```
## Not run: imputation <- mice(data = arg$data_mod,
                             method = arg$method,
                             predictorMatrix = arg$predictorMatrix,
                             JointModelEq=arg$JointModelEq,
                             control=arg$control,
                             maxit=1,m=5)

# Because of missing data only on one variable, fix maxit=1

# Estimation on each imputed dataset and pooling
analysis <- with(imputation,glm(hiv~age+condom+marital,family=binomial(link="probit")))
result <- pool(analysis)
summary(result)
## End(Not run)
```

MNARargument

Function providing modified arguments for imputation of Missing Not At Random (MNAR) outcomes using `mice()` function of the 'mice' package

Description

Imputation models for Missing Not At Random (MNAR) binary or continuous outcomes developed in this package use sample selection models. It is necessary, inside the imputation model, to specify a selection (i.e. missing data mechanism) and an outcome equation. The previous could be the model of interest (i.e. the post-imputation analysis model).

MNARargument adaptes `mice()` arguments:

1. data: Indicator of MNAR outcome missingness is included
2. method: For the MNAR outcome (varMNAR), MNAR imputation model is specified
3. predictorMatrix is modified to include MNAR indicator of missingness in other variable imputation model

Finally two new arguments are provided: JointModelEq, defining selection and outcome equation of the sample selection model; and control only for internal use.

The procedure is the following:

1. Use `generate_JointModelEq()` to construct an empty matrix of variable names allowing to specify selection and outcome equation
2. Fulfill the previous empty matrix adequately to selection and outcome equation specification of the sample selection model
3. Generate an object using `MNARargument()` function
4. Include in the `mice()` function the five arguments of the previous object generated by `MNARargument()`

Usage

```
MNARargument(data, method = NULL, predictorMatrix = NULL, varMNAR, JointModelEq = NULL)
```

Arguments

data	The dataset used for classical <code>mice()</code> and additional variables necessary for MNAR imputation models.
method	The <code>mice()</code> method argument.
predictorMatrix	The <code>mice()</code> predictorMatrix argument.
varMNAR	The name of MNAR outcome to be imputed.
JointModelEq	Matrix indicating variables included in selection and outcome equations of MNAR outcome imputation models.

Details

Be careful to not define the same selection and outcome equations for MNAR imputation models. A constraint of the sample selection model implies the inclusion of different sets of covariates, which may or not be nested in the selection equation and the outcome equation, to avoid collinearity issues. It has been recommended to include at least a supplementary variable in the selection equation. This variable should be known to be unlinked directly to the outcome.

Value

data_mod	Modified dataset including indicator of missingness for MNAR outcomes. Indicators of missingness are coded as "ind_" adding the name of MNAR outcomes.
method	Modified <code>mice()</code> method argument using <code>mice.impute.hecknorm()</code> and <code>mice.impute.heckprob()</code> as imputation methods respectively for continuous and binary outcomes.
predictorMatrix	Modified <code>mice()</code> predictorMatrix argument including indicator of MNAR outcomes missingness as predictors for MAR covariates.
JointModelEq	For internal use: Modified JointModelEq entry argument.
control	For internal use: MNAR outcomes.

Warning

This package is only validated for the imputation of MNAR outcome. However, it is implemented to impute several MNAR variables in the same process. Such implementation must be realised carefully.

Note

For MNAR continuous outcome, the Heckman's one step estimator is selected by default. However, the two-step estimator is available using `mice.impute.hecknorm2step()`. To use it, the method argument should be modified before inclusion in the `mice()` function.

Author(s)

Jacques-Emmanuel Galimard

References

Galimard, J.-E., Chevret, S., Protopopescu, C., and Resche-Rigon, M. (2016) A multiple imputation approach for MNAR mechanisms compatible with Heckman's model. *Statist. Med.*, 35: 2907-2920. doi:10.1002/sim.6902.

See Also

[mice](#) [copulaSampleSel](#) [SemiParBIV](#) [hiv](#) [selection](#) [generate_JointModelEq](#)

Examples

```
require(GJRM)
require(mvtnorm)
require(pbivnorm)
require(sampleSelection)

# Import dataset with a suspected MNAR mechanism
data("hiv")

# We select only one region (lusuka) and 5 variables
lusuka <- hiv[hiv$region==5,c("hiv", "age", "marital", "condom", "smoke")]

# Categorical variables have to be recoded as factor
lusuka$hiv <- as.factor(lusuka$hiv)

#####
#### Missing data only on a binary outcome ##
#####

# Specify a selection (missing data mechanism) and an outcome equation (analyse model)

# Generate an empty matrix

JointModelEq <- generate_JointModelEq(data=lusuka,varMNAR = "hiv")

# Fill in with 1 for variable included in equations
JointModelEq["hiv_var_sel"] <- c(0,1,1,1,1)
JointModelEq["hiv_var_out"] <- c(0,1,1,1,0)

# Generation of argument for MNAR imputation model in "mice()" function
arg <- MNARargument(data=lusuka,varMNAR="hiv",JointModelEq=JointModelEq)

# Imputation using mice() function
# Values returned have to be included in the "mice()" function as argument:

imputation <- mice(data = arg$data_mod,
                  method = arg$method,
                  predictorMatrix = arg$predictorMatrix,
                  JointModelEq=arg$JointModelEq,
                  control=arg$control,
                  maxit=1,m=5)
```

```

# Because of missing data only on one variable, fix maxit=1

# Estimation on each imputed dataset and pooling
analysis <- with(imputation,glm(hiv~age+condom+marital,family=binomial(link="probit")))
result <- pool(analysis)
summary(result)

#####
#### Missing data on a binary outcome and one covariate ##
#####

# Generate missing values on the variable "condom"
# According to a MAR mechanism using a probit model
prob <- pnorm((35.5-lusuka$age)/10.74) # Depending on "age"
lusuka$condom[rbinom(nrow(lusuka),size=1, prob=prob)==0] <- NA

JointModelEq <- generate_JointModelEq(data=lusuka,varMNAR = c("hiv"))
JointModelEq[, "hiv_var_sel"] <- c(0,1,1,1,1)
JointModelEq[, "hiv_var_out"] <- c(0,1,1,1,0)

arg <- MNARargument(data=lusuka,varMNAR=c("hiv"),JointModelEq=JointModelEq)

## Not run: # Imputation using mice function
imputation <- mice(data = arg$data_mod,
                  method = arg$method,
                  predictorMatrix = arg$predictorMatrix,
                  JointModelEq=arg$JointModelEq,
                  control=arg$control,
                  maxit=5,m=5)

# As classically, estimation on each imputed datasets and pooling
analysis <- with(imputation,glm(hiv~age+condom+marital,family=binomial(link="probit")))
result <- pool(analysis)
summary(result)
## End(Not run)

#####
#### Missing data only on a continuous outcome ##
#####

# Generation of a simulated dataset with MNAR mechanism on a continuous outcome

X1 <- rnorm(500,0,1)
X2 <- rbinom(500,1,0.5)
X3 <- rnorm(500,1,0.5)

errors <- rmvnorm(500,mean=c(0,0),sigma=matrix(c(1,0.3,0.3,1),nrow=2,byrow=TRUE))

Y <- X1+X2+errors[,1]
Ry <- ifelse(0.66+1*X1-0.5*X2+X3+errors[,2]>0,1,0)

Y[Ry==0] <- NA

```

```

simul_data <- data.frame(Y,X1,X2,X3)

JointModelEq <- generate_JointModelEq(data=simul_data,varMNAR = "Y")

JointModelEq[,"Y_var_sel"] <- c(0,1,1,1)
JointModelEq[,"Y_var_out"] <- c(0,1,1,0)

arg <- MNARargument(data=simul_data,varMNAR="Y",JointModelEq=JointModelEq)

imputation2 <- mice(data = arg$data_mod,
  method = arg$method,
  predictorMatrix = arg$predictorMatrix,
  JointModelEq=arg$JointModelEq,
  control=arg$control,
  maxit=1,m=5)

analysis2 <- with(imputation,lm(Y~X1+X2+X3))
result2 <- pool(analysis2)
summary(result2)

#####
## Using 2-step estimation ##
#####

arg <- MNARargument(data=simul_data,varMNAR="Y",JointModelEq=JointModelEq)
arg$method["Y"] <- "hecknorm2step"

## Not run: imputation3 <- mice(data = arg$data_mod,
  method = arg$method,
  predictorMatrix = arg$predictorMatrix,
  JointModelEq=arg$JointModelEq,
  control=arg$control,
  maxit=1,m=5)

analysis3 <- with(imputation3,lm(Y~X1+X2+X3))
result3 <- pool(analysis3)
summary(result3)
## End(Not run)

```

Index

`copulaSampleSel`, [3](#), [12](#)

`generate_JointModelEq`, [3](#), [10](#), [12](#)

`hiv`, [4](#), [12](#)

`mice`, [2](#), [3](#), [10–12](#)

`mice.impute.hecknorm`, [2](#), [4](#), [7](#), [11](#)

`mice.impute.hecknorm2step`, [6](#), [7](#), [11](#)

`mice.impute.heckprob`, [2](#), [8](#), [11](#)

`miceMNAR`, [4](#), [5](#), [7](#), [9](#)

`miceMNAR (miceMNAR-package)`, [2](#)

`miceMNAR-package`, [2](#)

`MNARargument`, [2](#), [5](#), [6](#), [8](#), [10](#)

`selection`, [3](#), [12](#)

`SemiParBIV`, [3](#), [12](#)