Package ‘ImputeRobust’

October 12, 2022

Type Package

Title Robust Multiple Imputation with Generalized Additive Models for Location Scale and Shape

Version 1.3-1

Date 2018-11-25

Author Daniel Salfran [aut, cre],
Martin Spieß [aut, ths]

Description Provides new imputation methods for the 'mice' package based on generalized additive models for location, scale, and shape (GAMLSS) as described in de Jong, van Buuren and Spiess <doi:10.1080/03610918.2014.911894>.

Maintainer Daniel Salfran <danielsalfran@gmail.com>

Depends gamlss, mice, R (>= 3.2.0)

Imports purrr, extremevalues, gamlss.dist, lattice

License GPL-3

LazyData true

RoxygenNote 6.1.1

Encoding UTF-8

NeedsCompilation no

Repository CRAN

Date/Publication 2018-11-30 12:10:03 UTC

R topics documented:

ImputeRobust-package *Multiple Imputation with Generalized Additive Models for Location, Scale, and Shape*

Description

De Jong (2012), De Jong, van Buuren and Spiess (2016) introduced a new imputation method based on generalized additive models for location, scale, and shape (Rigby and Stasinopoulos, 2005), which is a class of univariate regression models, where the assumption of an exponential family is relaxed and replaced by a general distribution family. This allows the a more flexible modelling than standard parametric imputation models of not only the location (e.g. the mean), but also the scale (e.g. variance), and the shape (e.g., skewness and kurtosis) of the conditional distribution of the dependent variable given all other variables.

Author(s)

Daniel Salfran <daniel.salfran@uni-hamburg.de>

Martin Spiess <martin.spiess@uni-hamburg.de>

References

de Jong, R., van Buuren, S. & Spiess, M. (2016) Multiple Imputation of Predictor Variables Using Generalized Additive Models. Communications in Statistics – Simulation and Computation, 45(3), 968–985.

de Jong, Roel. (2012). "Robust Multiple Imputation." Universität Hamburg. http://ediss.sub.uni-hamburg.de/volltexte/2012/5971/.

Rigby, R. A., and Stasinopoulos, D. M. (2005). Generalized Additive Models for Location, Scale and Shape. Journal of the Royal Statistical Society: Series C (Applied Statistics) 54 (3): 507–54.

ImpGamlssBootstrap *GAMLSS bootstrap method*

Description

Creates a random generation function for the missing values with bootstrap sample from the fitted GAMLSS model for the completely observed data.

Usage

```
ImpGamlssBootstrap(incomplete.data, fit, R, ...)
```

Arguments

`incomplete.data`
: Data frame with missings on one variable.

`fit`
: Random sample generator method.

`R`
: Boolean matrix with the response indicator.

`...`
: extra arguments for the control of the gamlss fitting function

Value

Returns a imputation sample generator.

ImpGamlssFit *GAMLSS imputation fit*

Description

This function takes a data set to fit a gamlss model and another to predict the expected parameters values. It returns a function that will generate a vector of random observations for the predicted parameters. The amount of random observations is the number of units on the dataset used to get such predictions.

Usage

```
ImpGamlssFit(data, new.data, family, n.ind.par, gam.mod,
  mod.planb = list(type = "pb", par = list(degree = 1, order = 1)),
  n.par.planb = n.ind.par, lin.terms = NULL, n.cyc = 5, bf.cyc = 5,
  cyc = 5, forceNormal = FALSE, trace = FALSE, ...)
```

Arguments

`data`
: Completely observed data frame to be used to fit a gamlss model estimate.

`new.data`
: Data frame used to predict the parameter values for some given right side x-values on the gamlss model.

`family`
: Family to be used for the response variable on the GAMLSS estimation.

`n.ind.par`
: Number of individual parameters to be fitted. Currently it only allows one or two because of stability issues for more parameters.

`gam.mod`
: list with the parameters of the GAMLSS imputation model.

`mod.planb`
: list with the parameters of the alternative GAMLSS imputation model.

`n.par.planb`
: number of individual parameters in the alternative model.

`lin.terms`
: Character vector specifying which (if any) predictor variables should enter the model linearly.

`n.cyc`
: number of cycles of the gamlss algorithm

`bf.cyc`
: number of cycles in the backfitting algorithm

`cyc`	number of cycles of the fitting algorithm
`forceNormal`	Flag that if set to 'TRUE' will use a normal family for the gamlss estimation as a last resource.
`trace`	whether to print at each iteration (TRUE) or not (FALSE)
`...`	extra arguments for the control of the gamlss fitting function

Value

Returns a method to generate random samples for the fitted gamlss model using "new.data" as covariates.

`mice.impute.gamlss` *Multiple Imputation with Generalized Additive Models for Location, Scale, and Shape.*

Description

Imputes univariate missing data using a generalized model for location, scale and shape.

Usage

```
mice.impute.gamlss(y, ry, x, family = NO, n.ind.par = 2,
  fitted.gam = NULL, gam.mod = list(type = "pb"), EV = TRUE, ...)

mice.impute.gamlssNO(y, ry, x, fitted.gam = NULL, EV = TRUE, ...)

mice.impute.gamlssBI(y, ry, x, fitted.gam = NULL, EV = TRUE, ...)

mice.impute.gamlssJSU(y, ry, x, fitted.gam = NULL, EV = TRUE, ...)

mice.impute.gamlssPO(y, ry, x, fitted.gam = NULL, EV = TRUE, ...)

mice.impute.gamlssTF(y, ry, x, fitted.gam = NULL, EV = TRUE, ...)

mice.impute.gamlssGA(y, ry, x, fitted.gam = NULL, EV = TRUE, ...)

mice.impute.gamlssZIBI(y, ry, x, fitted.gam = NULL, EV = TRUE, ...)

mice.impute.gamlssZIP(y, ry, x, fitted.gam = NULL, EV = TRUE, ...)

fit.gamlss(y, ry, x, family = NO, n.ind.par = 2, gam.mod = list(type
  = "pb"), ...)
```

Arguments

y	Numeric vector with incomplete data.
ry	Response pattern of 'y' ('TRUE'=observed, 'FALSE'=missing).
x	Design matrix with 'length(y)' rows and 'p' columns containing complete covariates.
family	Distribution family to be used by GAMLSS. It defaults to NO but a range of families can be defined by calling the corresponding "gamlssFAMILY" method.
n.ind.par	Number of parameters from the distribution family to be individually estimated.
fitted.gam	A predefined bootstrap gamlss method returned by `fit.gamlss`. Mice by default refits the model with each imputation. The parameter is here for a future faster modified mice function.
gam.mod	list with the parameters of the GAMLSS imputation model.
EV	Logical value to determine whether to correct or not extreme imputed values. This can arise due to too much flexibility of the gamlss model.
...	extra arguments for the control of the gamlss fitting function

Details

Imputation of `y` using generalized additive models for location, scale, and shape. A model is fitted with the observed part of the data set. Then a bootstrap sample is generated and used to refit the model and generate imputations.

The function `fit.gamlss` handles the fitting and the bootstrap and returns a method to generated imputations.

Being gamlss a flexible non parametric method, there may be problems with the fitting and imputation depending on the sample size. The imputation functions try to handle anomalies automatically, but results should be still inspected.

Value

Numeric vector with imputed values for missing `y` values

Author(s)

Daniel Salfran <daniel.salfran@uni-hamburg.de>

References

de Jong, R., van Buuren, S. & Spiess, M. (2016) Multiple Imputation of Predictor Variables Using Generalized Additive Models. Communications in Statistics – Simulation and Computation, 45(3), 968–985.

de Jong, Roel. (2012). "Robust Multiple Imputation." Universität Hamburg. http://ediss.sub.uni-hamburg.de/volltexte/2012/5971/.

Rigby, R. A., and Stasinopoulos, D. M. (2005). Generalized Additive Models for Location, Scale and Shape. Journal of the Royal Statistical Society: Series C (Applied Statistics) 54 (3): 507–54.

Examples

```
require(lattice)
# Create the imputed data sets

predMat <- matrix(rep(0,25), ncol = 5)
predMat[4,1] <- 1
predMat[4,5] <- 1
predMat[2,1] <- 1
predMat[2,5] <- 1
predMat[2,4] <- 1
predMat[3,1] <- 1
predMat[3,5] <- 1
predMat[3,4] <- 1
predMat[3,2] <- 1
imputed.sets <- mice(sample.data, m = 2,
                     method = c("", "gamlssPO",
                                "gamlss", "gamlssBI", ""),
                     visitSequence = "monotone",
                     predictorMatrix = predMat,
                     maxit = 1, seed = 973,
                     n.cyc = 1, bf.cyc = 1,
                     cyc = 1)

fit <- with(imputed.sets, lm(y ~ X.1 + X.2 + X.3 + X.4))
summary(pool(fit))

stripplot(imputed.sets)
```

`ModelCreator` *Model creator*

Description

This is a helper function to be used within the gamlss fitting procedure. It creates automatically a formula object for the variables named a given data frame. The dependent variable is the one in the first column and the rest are treated as independent.

Usage

```
ModelCreator(data, gam.model, lin.terms = NULL)
```

Arguments

`data`	Data frame that will provide the named variables.
`gam.model`	List of mode parameter, containing the "type" with c("linear", "cs", "pb") as available choices and "par", an optional list parameter if the model is not linear.
`lin.terms`	Specify which predictors should be included linearly. For example, binary variables can be added directly as an additive term instead of defining a spline.

Value

Returns a formula object.

`sample.data` *Sample data set with a monotone missing pattern*

Description

A simple data set with monotone missing pattern

Format

A data frame with 200 rows on the following 5 variables

- **X.1** Numeric variable from a Normal distribution
- **X.2** Count data from a Poisson distribution
- **X.3** Numeric variable from a Normal distribution
- **X.4** Binary variable from a Binomial distribution
- **y** Response variable

Details

Sample data set with four predictors and a dependent variable. A missing monotone pattern was generated in three predictors to illustrate the gamlss imputation method.

For the data generation process a parameter beta equal to `c(1.3, .8, 1.5, 2.5)` and a predictor matrix `X <-cbind(X.1, X.2, X.3, X.4)` are defined. Then, the sample data set is created with the model `y ~ X.1 + X.2 + X.3 + X.4`.

Examples

```
head(sample.data)
```

tao *Tropical Atmosphere Ocean (TAO) project data*

Description

A sample from the Tropical Atmosphere Ocean (TAO) project data, downloaded from the GGOBI project.

Format

A data frame with 736 observations on the following 8 variables.

Year a numeric vector

Latitude a numeric vector

Longitude a numeric vector

Sea.Surface.Temp a numeric vector

Air.Temp a numeric vector

Humidity a numeric vector

UWind a numeric vector

VWind a numeric vector

Details

All cases recorded for five locations and two time periods.

Source

https://github.com/ggobi/ggobi/blob/master/data/tao.csv

Examples

```
head(tao)
```

Index