

Package: midoc (via r-universe)

October 4, 2024

Title A Decision-Making System for Multiple Imputation

Version 1.0.0

Description A guidance system for analysis with missing data. It incorporates expert, up-to-date methodology to help researchers choose the most appropriate analysis approach when some data are missing. You provide the available data and the assumed causal structure, including the likely causes of missing data. 'midoc' will advise which analysis approaches can be used, and how best to perform them. 'midoc' follows the framework for the treatment and reporting of missing data in observational studies (TARMOS). Lee et al (2021).
<doi:10.1016/j.jclinepi.2021.01.008>.

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.1

Suggests knitr, shiny, testthat

Config/testthat/edition 3

Imports arm, blorr, dagitty, glue, grDevices, lifecycle, mfp2, mice
(>= 3.16.0), rlang, rmarkdown, stats, utils

VignetteBuilder knitr

Depends R (>= 3.6)

LazyData true

URL <https://elliecurnow.github.io/midoc/>

Repository <https://mrcieu.r-universe.dev>

RemoteUrl <https://github.com/elliecurnow/midoc>

RemoteRef HEAD

RemoteSha 2fb07f1112a31711e9749b036dd64966b9379178

Contents

bmi	2
checkCRA	3
checkMI	4
checkModSpec	5
descMissData	6
doMI mice	7
exploreDAG	8
midocVignette	9
proposeMI	9
Index	11

bmi	<i>Child body mass index data</i>
-----	-----------------------------------

Description

A simulated dataset

Usage

bmi

Format

bmi:

A data frame with 1000 rows and 6 columns:

bmi7 Child's body mass index at age 7 years

matage Mother's age at pregnancy, standardised relative to a mean age of 30

mated Mother's educational level: post-16 years qualification or not

pregsize Mother's pregnancy size: singleton or twins

bwt Child's birth weight in kilograms

r Missingness indicator: whether bmi7 is reported or not ...

`checkCRA`*Inspect complete records analysis model*

Description

Check complete records analysis is valid under the proposed analysis model and directed acyclic graph (DAG). Validity means that the proposed approach will allow unbiased estimation of the estimand(s) of interest, including regression parameters, associations, and causal effects.

Usage

```
checkCRA(y, covs, r_cra, mdag)
```

Arguments

<code>y</code>	The analysis model outcome, specified as a string
<code>covs</code>	The analysis model covariate(s), specified as a string (space delimited)
<code>r_cra</code>	The complete record indicator, specified as a string
<code>mdag</code>	The DAG, specified as a string using dagitty syntax

Details

The DAG should include all observed and unobserved variables related to the analysis model variables and their missingness, as well as all required missingness indicators.

In general, complete records analysis is valid if the analysis model outcome and complete record indicator are unrelated, conditional on the specified covariates. This is determined using the proposed DAG by checking whether the analysis model and complete record indicator are 'd-separated', given the covariates.

Value

A message indicating whether complete records analysis is valid under the proposed DAG and analysis model outcome and covariate(s)

References

Hughes R, Heron J, Sterne J, Tilling K. 2019. Accounting for missing data in statistical analyses: multiple imputation is not always the answer. *Int J Epidemiol*. doi:10.1093/ije/dyz032

Bartlett JW, Harel O, Carpenter JR. 2015. Asymptotically Unbiased Estimation of Exposure Odds Ratios in Complete Records Logistic Regression. *Am J Epidemiol*. doi:10.1093/aje/kwv114

Examples

```
# Example DAG for which complete records analysis is not valid, but could be
## valid for a different set of covariates
checkCRA(y="bmi7", covs="matage", r_cra="r",
         mdag="matage -> bmi7 mated -> matage mated -> bmi7
              sep_unmeas -> mated sep_unmeas -> r")
# For the DAG in the example above, complete records analysis is valid
## if a different set of covariates is used
checkCRA(y="bmi7", covs="matage mated", r_cra="r",
         mdag="matage -> bmi7 mated -> matage mated -> bmi7
              sep_unmeas -> mated sep_unmeas -> r")

# Example DAG for which complete records is not valid, but could be valid
## for a different estimand
checkCRA(y="bmi7", covs="matage mated", r_cra="r",
         mdag="matage -> bmi7 mated -> matage mated -> bmi7
              sep_unmeas -> mated sep_unmeas -> r matage -> bmi3
              mated -> bmi3 bmi3 -> bmi7 bmi3 -> r")

# Example DAG for which complete records analysis is never valid
checkCRA(y="bmi7", covs="matage mated", r_cra="r",
         mdag="matage -> bmi7 mated -> matage mated -> bmi7
              sep_unmeas -> mated sep_unmeas -> r bmi7 -> r")
```

checkMI

Inspect multiple imputation model

Description

Check multiple imputation is valid under the proposed imputation model and directed acyclic graph (DAG). Validity means that the proposed approach will allow unbiased estimation of the estimand(s) of interest, including regression parameters, associations, and causal effects. The imputation model should include all other analysis model variables as predictors, as well as any auxiliary variables. The DAG should include all observed and unobserved variables related to the analysis model variables and their missingness, as well as all required missingness indicators.

Usage

```
checkMI(dep, preds, r_dep, mdag)
```

Arguments

dep	The partially observed variable to be imputed, specified as a string
preds	The imputation model predictor(s), specified as a string (space delimited)
r_dep	The partially observed variable's missingness indicator, specified as a string
mdag	The DAG, specified as a string using dagitty syntax

Details

In principle, multiple imputation is valid if each partially observed variable is unrelated to its own missingness, given its imputation model predictors.

Value

A message indicating whether multiple imputation is valid under the proposed DAG and imputation model

References

Curnow E, Tilling K, Heron JE, Cornish RP, Carpenter JR. 2023. Multiple imputation of missing data under missing at random: including a collider as an auxiliary variable in the imputation model can induce bias. *Frontiers in Epidemiology*. doi:10.3389/fepid.2023.1237447

Examples

```
# Example DAG for which multiple imputation is valid
checkMI(dep="bmi7", preds="matage mated pregsz", r_dep="r",
        mdag="matage -> bmi7 mated -> matage mated -> bmi7
             sep_unmeas -> mated sep_unmeas -> r pregsz -> bmi7
             pregsz -> bwt sep_unmeas -> bwt")

# Example DAG for which multiple imputation is not valid, due to a collider
checkMI(dep="bmi7", preds="matage mated bwt", r_dep="r",
        mdag="matage -> bmi7 mated -> matage mated -> bmi7
             sep_unmeas -> mated sep_unmeas -> r pregsz -> bmi7
             pregsz -> bwt sep_unmeas -> bwt")
```

checkModSpec

Inspect parametric model specification

Description

Explore whether the observed relationships in the specified dataset are consistent with the proposed parametric model (which may represent the analysis or imputation model).

Usage

```
checkModSpec(formula, family, data, plot = TRUE, message = TRUE)
```

Arguments

formula A symbolic description of the model to be fitted, with the dependent variable on the left of a ~ operator, and the covariates, separated by + operators, on the right, specified as a string

family	A description of the error distribution and link function to be used in the model, specified as a string; family functions that are supported are "gaussian(identity)" and "binomial(logit)"
data	A data frame containing all the variables stated in the formula
plot	If TRUE (the default) and there is evidence of model mis-specification, displays a plot which can be used to explore the functional form of each covariate in the specified model; use plot = FALSE to disable the plot
message	If TRUE (the default), displays a message indicating whether the relationships between the dependent variable and covariates are likely to be correctly specified or not; use message = FALSE to suppress the message

Value

An object of type 'mimod' (a list containing the specified formula, family, and dataset name). Optionally, a message indicating whether the relationships between the dependent variable and covariates are likely to be correctly specified or not. If there is evidence of model mis-specification, optionally returns a plot of the model residuals versus the fitted values which can be used to explore the appropriate functional form for the specified model.

References

Curnow E, Carpenter JR, Heron JE, et al. 2023. Multiple imputation of missing data under missing at random: compatible imputation models are not sufficient to avoid bias if they are mis-specified. *J Clin Epidemiol*. doi:10.1016/j.jclinepi.2023.06.011

Examples

```
# Example (incorrectly) assuming a linear relationship
checkModSpec(formula="bmi7~matage+mated+pregsize",
              family="gaussian(identity)", data=bmi)
## For the example above, (correctly) assuming a quadratic relationship
checkModSpec(formula="bmi7~matage+I(matage^2)+mated+pregsize",
              family="gaussian(identity)", data=bmi)
```

descMissData

Lists missing data patterns in the specified dataset

Description

This function summarises the missing data patterns in the specified dataset. Each row in the output corresponds to a missing data pattern (1=observed, 0=missing). The number and percentage of observations is also displayed for each missing data pattern. The first column indicates the number of missing data patterns. The second column refers to the analysis model outcome ('y'), with all other variables ('covs') displayed in subsequent columns. Alternatively, 'y' can be used to display the primary variable of interest, e.g. 'y' could refer to the exposure, with all other variables listed in 'covs'.

Usage

```
descMissData(y, covs, data, plot = FALSE)
```

Arguments

y	The analysis model outcome, specified as a string
covs	The analysis model covariate(s), specified as a string (space delimited)
data	A data frame containing the specified analysis model outcome and covariate(s)
plot	If TRUE, displays a plot using md.pattern to visualise the missing data patterns; use plot = FALSE (the default) to disable the plot

Value

A summary of the missing data patterns

Examples

```
descMissData(y="bmi7", covs="matage mated", data=bmi)
descMissData(y="bmi7", covs="matage mated pregsz bwt", data=bmi, plot=TRUE)
```

doMImice

Performs multiple imputation

Description

Creates multiple imputations using [mice](#), based on the options and dataset specified by a call to [proposeMI](#). If a substantive model is specified, also calculates the pooled estimates using [pool](#).

Usage

```
doMImice(mipropobj, seed, substmod = " ", message = TRUE)
```

Arguments

mipropobj	An object of type 'miprop', created by a call to 'proposeMI'
seed	An integer that is used to set the seed of the 'mice' call
substmod	Optionally, a symbolic description of the substantive model to be fitted, specified as a string; if supplied, the model will be fitted to each imputed dataset and the results pooled
message	If TRUE (the default), displays a message summarising the analysis that has been performed; use message = FALSE to suppress the message

Value

A 'mice' object of class 'mids' (the multiply imputed datasets). Optionally, a message summarising the analysis that has been performed.

Examples

```
# First specify the imputation model as a 'mimod' object
## (suppressing the message)
mimod_bmi7 <- checkModSpec(formula="bmi7~matage+I(matage^2)+mated+pregsize",
                           family="gaussian(identity)",
                           data=bmi,
                           message=FALSE)

# Save the proposed 'mice' options as a 'miprop' object
## (suppressing the message)
miprop <- proposeMI(mimodobj=mimod_bmi7,
                   data=bmi,
                   message=FALSE,
                   plot = FALSE)

# Create the set of imputed datasets using the proposed 'mice' options
imp <- doMImice(miprop,123)

# Additionally, fit the substantive model to each imputed dataset and display
## the pooled results
doMImice(miprop, 123, substmod="lm(bmi7 ~ matage + I(matage^2) + mated)")
```

 exploreDAG

Compares data with proposed DAG

Description

Explore whether relationships between fully observed variables in the specified dataset are consistent with the proposed directed acyclic graph (DAG) using [localTests](#) functionality.

Usage

```
exploreDAG(mdag, data)
```

Arguments

mdag	The DAG, specified as a string using dagitty syntax
data	A data frame containing all the variables stated in the DAG. All ordinal variables must be integer-coded and all categorical variables must be dummy-coded.

Value

A message indicating whether the relationships between fully observed variables in the specified dataset are consistent with the proposed DAG

Examples

```
exploreDAG(mdag="matage -> bmi7 mated -> matage mated -> bmi7
             sep_unmeas -> mated sep_unmeas -> r",
           data=bmi)
```

midocVignette	<i>Run an interactive vignette for the midoc package</i>
---------------	--

Description

Runs an interactive version of the midoc vignette: Multiple Imputation DOctor (midoc). In the interactive version, you can apply midoc functions in [shiny-package](#) apps using your own DAG and data.

Usage

```
midocVignette()
```

Value

A browser-based, interactive version of the midoc vignette

Examples

```
# Run the interactive vignette
midocVignette()
```

proposeMI	<i>Suggests multiple imputation options</i>
-----------	---

Description

Suggests the [mice](#) options to perform multiple imputation, based on the proposed set of imputation models (one for each partially observed variable) and specified dataset.

Usage

```
proposeMI(mimodobj, data, plot = TRUE, plotprompt = TRUE, message = TRUE)
```

Arguments

mimodobj	An object, or list of objects, of type 'mimod', which stands for 'multiple imputation model', created by a call to checkModSpec
data	A data frame containing all the variables required for imputation and the substantive analysis
plot	If TRUE (the default), displays diagnostic plots for the proposed 'mice' call; use plot=FALSE to disable the plots
plotprompt	If TRUE (the default), the user is prompted before the second plot is displayed; use plotprompt=FALSE to remove the prompt
message	If TRUE (the default), displays a message describing the proposed 'mice' options; use message=FALSE to suppress the message

Value

An object of type 'miprop', which can be used to run 'mice' using the proposed options, plus, optionally, a message and diagnostic plots describing the proposed 'mice' options

Examples

```
# First specify each imputation model as a 'mimod' object
## (suppressing the message)
mimod_bmi7 <- checkModSpec(formula="bmi7~matage+I(matage^2)+mated+pregsize",
                           family="gaussian(identity)",
                           data=bmi,
                           message=FALSE)
mimod_pregsize <- checkModSpec(
  formula="pregsize~bmi7+matage+I(matage^2)+mated",
  family="binomial(logit)",
  data=bmi,
  message=FALSE)

# Display the proposed 'mice' options (suppressing the plot prompt)
## When specifying a single imputation model
proposeMI(mimodobj=mimod_bmi7,
          data=bmi,
          plotprompt = FALSE)
## When specifying more than one imputation model (suppressing the plots)
proposeMI(mimodobj=list(mimod_bmi7,mimod_pregsize),
          data=bmi,
          plot = FALSE)
```

Index

* **datasets**

bmi, [2](#)

bmi, [2](#)

checkCRA, [3](#)

checkMI, [4](#)

checkModSpec, [5](#), [9](#)

dagitty, [3](#), [4](#), [8](#)

descMissData, [6](#)

doMImice, [7](#)

exploreDAG, [8](#)

localTests, [8](#)

md.pattern, [7](#)

mice, [7](#), [9](#)

midocVignette, [9](#)

pool, [7](#)

proposeMI, [7](#), [9](#)

shiny-package, [9](#)