Package: ggmice (via r-universe)

November 2, 2026

Title Visualizations for 'mice' with 'ggplot2'

Version 0.1.2

Description Enhance a 'mice' imputation workflow with visualizations for incomplete and/or imputed data. The plotting functions produce 'ggplot' objects which may be easily manipulated or extended. Use 'ggmice' to inspect missing data, develop imputation models, evaluate algorithmic convergence, or compare observed versus imputed data.

License GPL (>= 3)

URL https://amices.org/ggmice/

BugReports https://github.com/amices/ggmice/issues

Imports cli, dplyr, ggplot2, magrittr, mice, purrr, rlang, scales, stats, stringr, tidyr, tidyselect, utils

Suggests covr, knitr, lifecycle, patchwork, plotly, rmarkdown, testthat (>= 3.0.0)

VignetteBuilder knitr

Config/testthat/edition 3

Copyright 'ggmice' authors

Encoding UTF-8

Roxygen list(markdown = TRUE)

Config/roxygen2/version 7.3.3.9000

RoxygenNote 7.3.3.9000

Config/pak/sysreqs cmake make libicu-dev libx11-dev zlib1g-dev

Repository https://amices.r-universe.dev

Date/Publication 2026-08-10 00:53:36 UTC

RemoteUrl https://github.com/amices/ggmice

RemoteRef HEAD

RemoteSha 92fac67ac82f3abc9760608b3ab45f5936743932

Contents

bwplot . . . 2
densityplot . . . 3
ggmice . . . 3
plot_corr . . . 5
plot_flux . . . 6
plot_miss . . . 7
plot_pattern . . . 8
plot_pred . . . 9
plot_trace . . . 11
stripplot . . . 12
xyplot . . . 12

Index **14**

`bwplot` *Box-and-whisker plot of observed and imputed data*

Description

Box-and-whisker plot of observed and imputed data

Usage

```
bwplot(...)
```

Arguments

`...`	Any arguments passed to the function.

Value

The output of mice::bwplot and a message about the `ggmice` equivalent.

Examples

```
imp <- mice::mice(mice::nhanes, maxit = 1, printFlag = FALSE)
bwplot(imp)
```

densityplot *Densityplot of observed and imputed data*

Description

Densityplot of observed and imputed data

Usage

```
densityplot(...)
```

Arguments

...	Any arguments passed to the function.

Value

The output of mice::densityplot and a message about the ggmice equivalent.

Examples

```
imp <- mice::mice(mice::nhanes, maxit = 1, printFlag = FALSE)
densityplot(imp)
```

ggmice *Plot incomplete or imputed data*

Description

Plot incomplete or imputed data

Usage

```
ggmice(data = NULL, mapping = ggplot2::aes())
```

Arguments

data	An incomplete dataset (of class data.frame), or an object of class mice::mids.
mapping	A list of aesthetic mappings created with ggplot2::aes().

Value

An object of class ggplot2::ggplot. The ggmice function returns output equivalent to ggplot2::ggplot output, with a few important exceptions:

- The theme is set to theme_mice.
- The color scale is set to the mice::mdc colors.
- The colour aesthetic is set to .where, an internally defined variable which distinguishes observed data from missing data or imputed data (for incomplete and imputed data, respectively).

See Also

See the ggmice vignette to use the ggmice() function on incomplete data or imputed data.

Examples

```
library(ggplot2)
# minimal example: scatterplot of incomplete and imputed data
dat <- mice::nhanes
ggmice(dat, aes(x = age, y = chl)) + geom_point()
imp <- mice::mice(dat, print = FALSE)
ggmice(imp, aes(x = age, y = chl)) + geom_point()

# more advanced functionality for incomplete data
# edit variable type for mixed incomplete data
dat$hyp <- factor(dat$hyp, levels = (1:2), labels = c("no hypertension", "hypertension"))
# scatterplot with categorical incomplete data
  ggmice(dat, aes(hyp, chl)) + geom_jitter(width = 0.1)
# incomplete data scatterplot faceted by categorical variable
ggmice(dat, aes(age, chl)) + geom_point() +
  facet_wrap(~ hyp, labeller = label_both)
# incomplete data scatterplot faceted by missing data indicator
ggmice(dat, aes(age, chl)) + geom_point() +
  facet_wrap(~ factor(is.na(hyp) == 0, labels = c("hyp observed", "hyp missing")))

# more advanced functionality for imputed data
# stripplot by imputation
ggmice(imp, aes(x = .imp, y = chl)) + geom_jitter(width = 0.25) +
  labs(x = "Imputation number")
# box plot by imputation
ggmice(imp, aes(x = .imp, y = chl)) + geom_boxplot() +
  labs(x = "Imputation number")
# density plot by imputation
ggmice(imp, aes(x = chl, group = .imp)) + geom_density()
# scatterplot faceted by imputation number
ggmice(imp, ggplot2::aes(x = age, y = bmi)) + ggplot2::geom_point() +
  facet_wrap(~ .imp)
```

plot_corr *Plot correlations between (incomplete) variables*

Description

Plot correlations between (incomplete) variables

Usage

```
plot_corr(
  data,
  vrb = "all",
  label = FALSE,
  square = TRUE,
  diagonal = FALSE,
  rotate = FALSE,
  caption = TRUE
)
```

Arguments

data	A dataset of class data.frame, tibble, or matrix.
vrb	String, vector, or unquoted expression with variable name(s), default is "all".
label	Logical indicating whether correlation values should be displayed.
square	Logical indicating whether the plot tiles should be squares.
diagonal	Logical indicating whether the correlation of each variable with itself should be displayed.
rotate	Logical indicating whether the variable name labels should be rotated 90 degrees.
caption	Logical indicating whether the figure caption should be displayed.

Value

An object of class ggplot2::ggplot.

Examples

```
# plot correlations for all columns
plot_corr(mice::nhanes)

# plot correlations for specific columns by supplying a character vector
plot_corr(mice::nhanes, c("chl", "hyp"))

# plot correlations for specific columns by supplying unquoted variable names
plot_corr(mice::nhanes, c(chl, hyp))
```

```
# plot correlations for specific columns by passing an object with variable names
# from the environment, unquoted with `!!`
my_variables <- c("chl", "hyp")
plot_corr(mice::nhanes, !!my_variables)
# object with variable names must be unquoted with `!!`
try(plot_corr(mice::nhanes, my_variables))
```

`plot_flux` *Plot the influx and outflux of a multivariate missing data pattern*

Description

Plot the influx and outflux of a multivariate missing data pattern

Usage

```
plot_flux(data, vrb = "all", jitter = TRUE, label = TRUE, caption = TRUE)
```

Arguments

`data`	An incomplete dataset of class `data.frame` or `matrix`.
`vrb`	String, vector, or unquoted expression with variable name(s), default is "all".
`jitter`	Logical indicating whether some noise is added to every point to prevent over-plotting (the default).
`label`	Logical indicating whether variable names should be displayed within the plot (the default) or with colors in the legend.
`caption`	Logical indicating whether the figure caption should be displayed.

Value

An object of class ggplot2::ggplot.

Examples

```
# plot flux for all columns
plot_flux(mice::nhanes)

# plot flux for specific columns by supplying a character vector
plot_flux(mice::nhanes, c("chl", "hyp"))

# plot flux for specific columns by supplying unquoted variable names
plot_flux(mice::nhanes, c(chl, hyp))

# plot flux for specific columns by passing an object with variable names
# from the environment, unquoted with `!!`
my_variables <- c("chl", "hyp")
plot_flux(mice::nhanes, !!my_variables)
```

```
# object with variable names must be unquoted with `!!`
try(plot_flux(mice::nhanes, my_variables))
```

---

plot_miss *Plot missingness in a dataset*

Description

[Experimental]

Usage

```
plot_miss(
  data,
  vrb = "all",
  ordered = FALSE,
  rotate = FALSE,
  grid = FALSE,
  square = FALSE
)
```

Arguments

data	An incomplete dataset of class data.frame or matrix.
vrb	String, vector, or unquoted expression with variable name(s), default is "all".
ordered	Logical indicating whether rows should be ordered according to their pattern.
rotate	Logical indicating whether the variable name labels should be rotated 90 degrees.
grid	Logical indicating whether borders should be present between tiles.
square	Logical indicating whether the plot tiles should be squares, defaults to squares.

Value

An object of class ggplot2::ggplot.

Examples

```
# plot correlations for all columns
plot_miss(mice::nhanes)

# plot correlations for specific columns by supplying a character vector
plot_miss(mice::nhanes, c("chl", "hyp"))

# plot correlations for specific columns by supplying unquoted variable names
plot_miss(mice::nhanes, c(chl, hyp))
```

```
# plot correlations for specific columns by passing an object with variable names
# from the environment, unquoted with `!!`
my_variables <- c("chl", "hyp")
plot_miss(mice::nhanes, !!my_variables)
# object with variable names must be unquoted with `!!`
try(plot_miss(mice::nhanes, my_variables))

# plot larger dataset
plot_miss(mice::boys)
plot_miss(mice::boys, ordered = TRUE)
```

`plot_pattern` *Plot the missing data pattern of an incomplete dataset*

Description

Plot the missing data pattern of an incomplete dataset

Usage

```
plot_pattern(
  data,
  vrb = "all",
  square = TRUE,
  rotate = FALSE,
  cluster = NULL,
  npat = NULL,
  caption = TRUE
)
```

Arguments

`data`	An incomplete dataset of class `data.frame` or `matrix`.
`vrb`	String, vector, or unquoted expression with variable name(s), default is "all".
`square`	Logical indicating whether the plot tiles should be squares, defaults to squares to mimic `mice::md.pattern()`.
`rotate`	Logical indicating whether the variable name labels should be rotated 90 degrees.
`cluster`	Optional character string specifying which variable should be used for clustering (e.g., for multilevel data).
`npat`	Optional numeric input specifying the maximum number of missing data patterns to be visualized, defaults to all patterns.
`caption`	Logical indicating whether the figure caption should be displayed.

Value

An object of class ggplot2::ggplot.

Examples

```
# plot missing data pattern for all columns
plot_pattern(mice::nhanes)

# plot missing data pattern for specific columns by supplying a character vector
plot_pattern(mice::nhanes, c("chl", "hyp"))

# plot missing data pattern for specific columns by supplying unquoted variable names
plot_pattern(mice::nhanes, c(chl, hyp))

# plot missing data pattern for specific columns by passing an object with variable names
# from the environment, unquoted with `!!`
my_variables <- c("chl", "hyp")
plot_pattern(mice::nhanes, !!my_variables)
# object with variable names must be unquoted with `!!`
try(plot_pattern(mice::nhanes, my_variables))

# plot missing data pattern by cluster
plot_pattern(mice::nhanes, cluster = "age")
```

---

plot_pred *Plot the predictor matrix of an imputation model*

Description

Plot the predictor matrix of an imputation model

Usage

```
plot_pred(
  data,
  vrb = "all",
  method = NULL,
  label = TRUE,
  square = TRUE,
  rotate = FALSE
)
```

Arguments

data	A predictor matrix for mice, typically generated with mice::make.predictorMatrix or mice::quickpred, or an object of class mice::mids.
vrb	String, vector, or unquoted expression with variable name(s), default is "all".

method	Character string or vector with imputation methods.
label	Logical indicating whether predictor matrix values should be displayed.
square	Logical indicating whether the plot tiles should be squares.
rotate	Logical indicating whether the variable name labels should be rotated 90 degrees.

Details

The predictor matrix in mice::mice determines the role an imputation model predictor takes in the imputation model. The rows correspond to incomplete target variables, and the columns to imputation model predictors.

A value of 1 indicates that the column variable is a predictor to impute the target (row) variable. The value 0 means that it is not used as predictor.

Imputation methods for multilevel data use other codes than 0 and 1:

- Methods 2l.bin, 2l.lmer, 2l.norm, 2l.pan, 2lonly.mean, 2lonly.norm and 2lonly.pmm use code -2 to indicate the class variable;
- Methods 2l.bin, 2l.lmer, 2l.norm and 2l.pan use code 2 to indicate the random effects;
- Method 2l.pan uses codes 3 and 4 to add class means to codes 1 and 2 respectively.

Value

An object of class ggplot2::ggplot.

References

van Buuren, S. (2018). Flexible imputation of missing data. Chapman and Hall/CRC. stefvanbuuren.name/fimd

Examples

```
# generate a predictor matrix
pred <- mice::quickpred(mice::nhanes)

# plot predictor matrix for all columns
plot_pred(pred)

# plot predictor matrix for specific columns by supplying a character vector
plot_pred(pred, c("chl", "hyp"))

# plot predictor matrix for specific columns by supplying unquoted variable names
plot_pred(pred, c(chl, hyp))

# plot predictor matrix for specific columns by passing an object with variable names
# from the environment, unquoted with `!!`
my_variables <- c("chl", "hyp")
plot_pred(pred, !!my_variables)
# object with variable names must be unquoted with `!!`
try(plot_pred(pred, my_variables))
```

```
# plot predictor matrix of mids object
imp <- mice::mice(mice::nhanes, print = FALSE)
plot_pred(imp)
```

`plot_trace` *Plot the trace lines of the imputation algorithm*

Description

Plot the trace lines of the imputation algorithm

Usage

```
plot_trace(data, vrb = "all", trend = FALSE, legend = TRUE)
```

Arguments

`data`	An object of class mice::mids.
`vrb`	String, vector, or unquoted expression with variable name(s), default is "all".
`trend`	Logical indicating whether a smoothened trend should be added, default is FALSE.
`legend`	Logical indicating whether the plot legend should be visible, default is TRUE.

Details

The `vrb` argument is "quoted" via `rlang::enexpr()` and evaluated according to tidy evaluation principles. In practice, this technical nuance only affects users when passing an object from the environment (e.g., a vector of variable names) to the `vrb` argument. In such cases, the object must be "unquoted" via the `!!` prefix operator.

Value

An object of class ggplot2::ggplot.

Examples

```
# create [mice::mids] object with [mice::mice()]
imp <- mice::mice(mice::nhanes, print = FALSE)

# plot trace lines for all imputed columns
plot_trace(imp)

# plot trace lines for specific columns by supplying a string or character vector
plot_trace(imp, "chl")
plot_trace(imp, c("chl", "hyp"))
# plot trace lines for specific columns by supplying unquoted variable names
plot_trace(imp, chl)
```

```
plot_trace(imp, c(chl, hyp))

# plot trace lines for specific columns by passing an object with variable names
# from the environment, unquoted with `!!`
my_variables <- c("chl", "hyp")
plot_trace(imp, !!my_variables)
# object with variable names must be unquoted with `!!`
try(plot_trace(imp, my_variables))
```

`stripplot` *Stripplot of observed and imputed data*

Description

Stripplot of observed and imputed data

Usage

```
stripplot(...)
```

Arguments

`...`	Any arguments passed to the function.

Value

The output of mice::stripplot and a message about the `ggmice` equivalent.

Examples

```
imp <- mice::mice(mice::nhanes, maxit = 1, printFlag = FALSE)
stripplot(imp)
```

`xyplot` *Scatterplot of observed and imputed data*

Description

Scatterplot of observed and imputed data

Usage

```
xyplot(...)
```

Arguments

`...`	Any arguments passed to the function.

Value

The output of mice::xyplot and a message about the ggmice equivalent.

Examples

```
imp <- mice::mice(mice::nhanes, maxit = 1, printFlag = FALSE)
xyplot(imp, bmi ~ age)
```

Index

bwplot, 2

densityplot, 3

ggmice, 3, *4*
ggplot2::aes(), *3*
ggplot2::ggplot, *4–7*, *9*, *11*

mice::bwplot, *2*
mice::densityplot, *3*
mice::make.predictorMatrix, *9*
mice::mdc, *4*
mice::mice, *10*
mice::mids, *3*, *9*, *11*
mice::quickpred, *9*
mice::stripplot, *12*
mice::xyplot, *13*

plot_corr, 5
plot_flux, 6
plot_miss, 7
plot_pattern, 8
plot_pred, 9
plot_trace, 11

rlang::enexpr(), *11*

stripplot, 12

theme_mice, *4*

xyplot, 12