Package 'broomExtra'

April 3, 2019

Type Package

Title Enhancements for 'broom' Package Family

Version 0.0.2

Maintainer Indrajeet Patil <patilindrajeet.science@gmail.com>

Description Collection of functions to assist 'broom' and
'broom.mixed' package-related data analysis workflows. In particular,
the generic functions tidy(), glance(), and augment() choose
appropriate S3 methods from these two packages depending on which
package exports the needed method. Additionally, 'grouped_' variants
of the generics provides a convenient way to execute functions across
a combination of grouping variable(s) in a dataframe.

License GPL-3 | file LICENSE

URL https://indrajeetpatil.github.io/broomExtra/,
https://github.com/IndrajeetPatil/broomExtra

BugReports https://github.com/IndrajeetPatil/broomExtra/issues

Depends R (>= 3.5.0)

Imports broom (>= 0.5.1),
broom.mixed (>= 0.2.4),
dplyr (>= 0.8.0.1),
magrittr (>= 1.5),
rlang (>= 0.3.1)

Suggests covr,
gapminder,
generics,
ggplot2,
knitr,
lme4,
rmarkdown,
spelling,
stringr,
testthat,
tibble,
tidyr,
utils

VignetteBuilder knitr

Encoding UTF-8

Language en-US

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 6.1.1

R topics documented:

augment	*Retrieve augmented dataframe if it exists.*

Description

Check if a augment method exits for a given object, either in broom or in broom.mixed. If it does, return the model summary dataframe, if not, return a NULL.

Usage

```
augment(x, ...)
```

Arguments

x	Model object or other R object with information to append to observations.
...	Addition arguments to augment method.

Value

A tibble::tibble() with information about data points.

Methods

No methods found in currently loaded packages.

Author(s)

Indrajeet Patil

Examples

```
set.seed(123)
library(lme4)

# mixed-effects models (`broom.mixed` will be used)
lmm.mod <- lmer(Reaction ~ Days + (Days | Subject), sleepstudy)
broomExtra::augment(lmm.mod)

# linear model (`broom` will be used)
lm.mod <- lm(Reaction ~ Days, sleepstudy)
broomExtra::augment(lm.mod)
```

---

glance	*Retrieve model summary dataframe if it exists.*

Description

Check if a `glance` method exits for a given object, either in `broom` or in `broom.mixed`. If it does, return the model summary dataframe, if not, return a `NULL`.

Usage

```
glance(x, ...)
```

Arguments

x	model or other R object to convert to single-row data frame
...	other arguments passed to methods

Methods

No methods found in currently loaded packages.

Author(s)

Indrajeet Patil

Examples

```
set.seed(123)
library(lme4)

# mixed-effects models (`broom.mixed` will be used)
lmm.mod <- lmer(Reaction ~ Days + (Days | Subject), sleepstudy)
broomExtra::glance(lmm.mod)

# linear model (`broom` will be used)
lm.mod <- lm(Reaction ~ Days, sleepstudy)
broomExtra::glance(lm.mod)
```

grouped_augment	*Augmented data from grouped analysis of any function that has* data *argument in its function call.*

Description

Augmented data from grouped analysis of any function that has data argument in its function call.

Usage

```
grouped_augment(data, grouping.vars, ..f, ..., augment.args = list())
```

Arguments

data	Dataframe (or tibble) from which variables are to be taken.
grouping.vars	Grouping variables.
..f	A function, or function name as a string.
...	Arguments to function.
	These dots support [tidy-dots](#) features.
augment.args	A list of arguments to be used in the relevant S3 method.

Value

A [tibble::tibble()](#) with information about data points.

Methods

No methods found in currently loaded packages.

Author(s)

Indrajeet Patil

Examples

```
set.seed(123)
# to speed up computation, let's use only 50% of the data

# linear model
broomExtra::grouped_augment(
  data = dplyr::sample_frac(tbl = ggplot2::diamonds, size = 0.5),
  grouping.vars = c(cut, color),
  formula = price ~ carat - 1,
  ..f = stats::lm,
  na.action = na.omit
)

# linear mixed effects model
broomExtra::grouped_augment(
  data = dplyr::sample_frac(tbl = ggplot2::diamonds, size = 0.5),
  grouping.vars = cut,
  ..f = lme4::lmer,
```

```
    formula = price ~ carat + (carat | color) - 1,
    control = lme4::lmerControl(optimizer = "bobyqa")
)
```

---

grouped_glance	*Model summary output from grouped analysis of any function that has* data *argument in its function call.*

Description

Model summary output from grouped analysis of any function that has data argument in its function call.

Usage

```
grouped_glance(data, grouping.vars, ..f, ...)
```

Arguments

data	Dataframe (or tibble) from which variables are to be taken.
grouping.vars	Grouping variables.
..f	A function, or function name as a string.
...	Arguments to function.
	These dots support tidy-dots features.

Methods

No methods found in currently loaded packages.

Author(s)

Indrajeet Patil

Examples

```
set.seed(123)
# to speed up computation, let's use only 50% of the data

# linear model
broomExtra::grouped_glance(
  data = dplyr::sample_frac(tbl = ggplot2::diamonds, size = 0.5),
  grouping.vars = c(cut, color),
  formula = price ~ carat - 1,
  ..f = stats::lm,
  na.action = na.omit
)

# linear mixed effects model
broomExtra::grouped_glance(
  data = dplyr::sample_frac(tbl = ggplot2::diamonds, size = 0.5),
  grouping.vars = cut,
  ..f = lme4::lmer,
```

```
    formula = price ~ carat + (carat | color) - 1,
    control = lme4::lmerControl(optimizer = "bobyqa")
)
```

---

grouped_tidy   *Tidy output from grouped analysis of any function that has* data *argu-*
                             ment in its function call.

Description

Tidy output from grouped analysis of any function that has data argument in its function call.

Usage

```
grouped_tidy(data, grouping.vars, ..f, ..., tidy.args = list())
```

Arguments

data	Dataframe (or tibble) from which variables are to be taken.
grouping.vars	Grouping variables.
..f	A function, or function name as a string.
...	Arguments to function.
	These dots support tidy-dots features.
tidy.args	A list of arguments to be used in the relevant S3 method.

Value

A tibble::tibble() with information about model components.

Methods

No methods found in currently loaded packages.

Author(s)

Indrajeet Patil

Examples

```
set.seed(123)
# to speed up computation, let's use only 50% of the data

# linear model
broomExtra::grouped_tidy(
  data = dplyr::sample_frac(tbl = ggplot2::diamonds, size = 0.5),
  grouping.vars = c(cut, color),
  formula = price ~ carat - 1,
  ..f = stats::lm,
  na.action = na.omit,
  tidy.args = list(quick = TRUE)
)
```

```
# linear mixed effects model
broomExtra::grouped_tidy(
  data = dplyr::sample_frac(tbl = ggplot2::diamonds, size = 0.5),
  grouping.vars = cut,
  ..f = lme4::lmer,
  formula = price ~ carat + (carat | color) - 1,
  control = lme4::lmerControl(optimizer = "bobyqa"),
  tidy.args = list(conf.int = TRUE, conf.level = 0.99)
)
```

tidy   *Retrieve tidy dataframe if it exists.*

Description

Checks if a `tidy` method exits for a given object, either in `broom` or in `broom.mixed`. If it does, it turn an object into a tidy tibble, if not, return a `NULL`.

Usage

```
tidy(x, ...)
```

Arguments

x	An object to be converted into a tidy [tibble::tibble()]().
...	Additional arguments to tidying method.

Value

A [tibble::tibble()]() with information about model components.

Methods

No methods found in currently loaded packages.

Author(s)

Indrajeet Patil

Examples

```
set.seed(123)
library(lme4)

# mixed-effects models (`broom.mixed` will be used)
lmm.mod <- lmer(Reaction ~ Days + (Days | Subject), sleepstudy)
broomExtra::tidy(lmm.mod, effects = "fixed", exponentiate = TRUE)

# linear model (`broom` will be used)
lm.mod <- lm(Reaction ~ Days, sleepstudy)
broomExtra::tidy(lm.mod, conf.int = TRUE)
```

Index