Package 'broomExtra'

October 26, 2019

Type Package

Title Enhancements for 'broom' Package Family

Version 0.0.6

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_' and
'boot_' variants of the generics provides a convenient way to execute
functions across a combination of grouping variable(s) in a dataframe
or bootstrap them.

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 (>= 0.5.2),
broom.mixed (>= 0.2.4),
dplyr (>= 0.8.3),
magrittr (>= 1.5),
purrr (>= 0.3.3),
rlang (>= 0.4.1),
rsample (>= 0.0.5),
tidyr (>= 1.0.0)

Suggests generics,
ggplot2,
knitr,
lme4,
rmarkdown,
spelling,
stringr,
testthat

VignetteBuilder knitr

Encoding UTF-8

Language en-US

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 6.1.99.9001

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.

Note

For available methods, see- [`https://indrajeetpatil.github.io/broomExtra/articles/available_methods.html`](https://indrajeetpatil.github.io/broomExtra/articles/available_methods.html)

Author(s)

Indrajeet Patil

See Also

grouped_augment, boot_augment

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)
```

---

boot_augment   *Bootstrapped dataframe with augmented predictions from each sample.*

Description

Bootstrapped dataframe with augmented predictions from each sample.

Usage

```
boot_augment(
  data,
  times = 25,
  strata = NULL,
  apparent = FALSE,
  ..f,
  ...,
  augment.args = list()
)
```

Arguments

data	Dataframe (or tibble) from which variables are to be taken.
times	The number of bootstrap samples.
strata	A variable that is used to conduct stratified sampling. When not NULL, each bootstrap sample is created within the stratification variable. This could be a single character value or a variable name that corresponds to a variable that exists in the data frame.
apparent	A logical. Should an extra resample be added where the analysis and holdout subset are the entire data set. This is required for some estimators used by the summary function that require the apparent error rate.

..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.

Note

For available methods, see- https://indrajeetpatil.github.io/broomExtra/articles/available_methods.html

Author(s)

Indrajeet Patil

See Also

grouped_augment, augment

Examples

```
set.seed(123)

# example-1: linear model
broomExtra::boot_augment(
  data = mtcars,
  times = 10,
  ..f = stats::lm,
  formula = mpg ~ wt,
  na.action = na.omit
)

# example-2: linear mixed-effects model
library(lme4)

broomExtra::boot_augment(
  data = sleepstudy,
  times = 25,
  ..f = lme4::lmer,
  formula = Reaction ~ Days + (Days | Subject)
)
```

---

boot_glance   *Bootstrapped dataframe with model summaries from each sample.*

Description

Bootstrapped dataframe with model summaries from each sample.

Usage

```
boot_glance(data, times = 25, strata = NULL, apparent = FALSE, ..f, ...)
```

Arguments

data	Dataframe (or tibble) from which variables are to be taken.
times	The number of bootstrap samples.
strata	A variable that is used to conduct stratified sampling. When not NULL, each bootstrap sample is created within the stratification variable. This could be a single character value or a variable name that corresponds to a variable that exists in the data frame.
apparent	A logical. Should an extra resample be added where the analysis and holdout subset are the entire data set. This is required for some estimators used by the summary function that require the apparent error rate.
..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.

Note

For available methods, see- `https://indrajeetpatil.github.io/broomExtra/articles/available_methods.html`

Author(s)

[Indrajeet Patil](#)

See Also

[grouped_glance](#), [glance](#)

Examples

```
set.seed(123)

# example-1: linear model
broomExtra::boot_glance(
  data = mtcars,
  times = 500,
  ..f = stats::lm,
  formula = mpg ~ wt,
  na.action = na.omit
)

# example-2: linear mixed-effects model
library(lme4)

broomExtra::boot_glance(
  data = sleepstudy,
  times = 25,
  ..f = lme4::lmer,
  formula = Reaction ~ Days + (Days | Subject)
)
```

---

boot_tidy   *Bootstrapped dataframe with estimates from each sample.*

Description

Bootstrapped dataframe with estimates from each sample.

Usage

```
boot_tidy(
  data,
  times = 25,
  strata = NULL,
  apparent = FALSE,
  ..f,
  ...,
  tidy.args = list()
)
```

Arguments

data	Dataframe (or tibble) from which variables are to be taken.
times	The number of bootstrap samples.
strata	A variable that is used to conduct stratified sampling. When not NULL, each bootstrap sample is created within the stratification variable. This could be a single character value or a variable name that corresponds to a variable that exists in the data frame.
apparent	A logical. Should an extra resample be added where the analysis and holdout subset are the entire data set. This is required for some estimators used by the summary function that require the apparent error rate.

..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.

Note

For available methods, see- `https://indrajeetpatil.github.io/broomExtra/articles/available_methods.html`

Author(s)

Indrajeet Patil

See Also

`grouped_tidy`, `tidy`

Examples

```
set.seed(123)

# example-1: linear model
broomExtra::boot_tidy(
  data = mtcars,
  times = 500,
  ..f = stats::lm,
  formula = mpg ~ wt,
  na.action = na.omit,
  tidy.args = list(conf.int = TRUE, conf.level = 0.50)
)

# example-2: linear mixed-effects model
library(lme4)

broomExtra::boot_tidy(
  data = sleepstudy,
  times = 25,
  ..f = lme4::lmer,
  formula = Reaction ~ Days + (Days | Subject),
  tidy.args = list(effects = "fixed")
)
```

---

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.

Note

For available methods, see- https://indrajeetpatil.github.io/broomExtra/articles/available_methods.html

Author(s)

Indrajeet Patil

See Also

grouped_glance, boot_glance

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.

Note

For available methods, see- https://indrajeetpatil.github.io/broomExtra/articles/available_methods.html

Author(s)

[Indrajeet Patil](#)

See Also

[augment](#), [boot_augment](#)

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,
  augment.args = list(se_fit = TRUE)
)

# 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.

Note

For available methods, see- https://indrajeetpatil.github.io/broomExtra/articles/available_
methods.html

Author(s)

Indrajeet Patil

See Also

glance, boot_glance

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 *argument 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.

Note

For available methods, see- https://indrajeetpatil.github.io/broomExtra/articles/available_methods.html

Author(s)

Indrajeet Patil

See Also

tidy, boot_tidy

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`. In case of data frames, a tibble data frame is returned.

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.

Note

For available methods, see- https://indrajeetpatil.github.io/broomExtra/articles/available_methods.html

Author(s)

[Indrajeet Patil](#)

See Also

[grouped_tidy](#), [boot_tidy](#)

Examples

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

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

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

# unsupported object (the function will return `NULL` in such cases)
broomExtra::tidy(list(1, c("x", "y")))
```

Index