Package ‘generics’

November 29, 2018

Version 0.0.2

Title Common S3 Generics not Provided by Base R Methods Related to Model Fitting

Description In order to reduce potential package dependencies and conflicts, generics provides a number of commonly used S3 generics.

License GPL-2

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 6.1.1

URL https://github.com/r-lib/generics

BugReports https://github.com/r-lib/generics

Depends R (>= 3.1)

Suggests covr,
    pkgload,
    testthat,
    tibble

Imports methods

R topics documented:

  augment ................................................................. 2
  calculate ............................................................. 2
  coercion-factor ..................................................... 3
  coercion-time-difference ........................................... 4
  compile ............................................................... 5
  components ............................................................. 5
  equation ............................................................... 6
  estfun ................................................................. 6
  evaluate ............................................................... 7
  explain ................................................................. 7
  fit ...................................................................... 8
  fit_xy ................................................................. 8
  generate .............................................................. 9
  glance ................................................................. 9
  hypothesize ............................................................ 10
**augment**

*Augment data with information from an object*

**Description**

Augment data with information from an object

**Usage**

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

**Arguments**

- `x` Model object or other R object with information to append to observations.
- `...` Additional arguments to `augment` method.

**Value**

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

**Methods**

No methods found in currently loaded packages.

**calculate**

*Calculate statistics.*

**Description**

Calculate statistics.

**Usage**

```
calculate(x, ...)
```
Arguments

- `x` An object.
- `...` Other arguments passed to methods

Methods

No methods found in currently loaded packages.

---

### Description

Coercion functions for creating factors from other existing objects.

### Usage

- `as.factor(x, ...)`
- `as.ordered(x, ...)`

### Arguments

- `x` A vector of data.
- `...` Other arguments passed on to methods.

### Details

These functions override non-generic factor coercion functions provided in base so that packages can provide methods for different data types. The default methods call the base versions.

### Value

For `as.factor()`, a factor. For `as.ordered()`, an ordered factor.

### Methods

- `as.factor()`: No methods found in currently loaded packages.
- `as.ordered()`: No methods found in currently loaded packages.

### Examples

- `as.factor(letters[1:5])`
- `as.ordered(letters[1:5])`
Description

Coercion functions for creating `difftime` objects from other existing objects.

Usage

```r
as.difftime(tim, ...)  
## Default S3 method:  
as.difftime(tim, format = "%X", units = "auto", ...)
```

Arguments

- `tim`: A vector specifying a time interval.
- `...`: Other arguments passed on to methods.
- `format`: A single character specifying the format of `tim` when it is a character. The default is a locale-specific time format.
- `units`: A single character specifying units in which the results are desired. Required if `tim` is a numeric.

Details

This function overrides the non-generic `as.difftime()` function provided in base so that packages can provide methods for different data types. The default method call the base version.

Value

A `difftime` object with an attribute indicating the units.

Methods

See the following help topics for more details about individual methods:

- generics

  - `coercion-time-difference`: default

Examples

```r
as.difftime(1:5, units = "secs")  
as.difftime(c("01:55:22", "01:55:25"))  
as.difftime("01", format = "%H")  
as.difftime("01", format = "%H", units = "secs")
compile

Compile an object

Description
Finalizes or completes an object.

Usage
compile(object, ...)

Arguments
object An object. See the individual method for specifics.
... Other arguments passed to methods

Methods
No methods found in currently loaded packages.

components

Extract components

Description
components can be used to extract elements from an object.

Usage
components(object, ...)

Arguments
object A data separable object.
... Other arguments passed to methods

Details
For example, decomposition methods and some modelling techniques can be used to decompose a dataset into components of interest. This function is used to extract these components in a tidy data format.

Value
A dataset (tibble::tibble) or similar) containing components from the object.

Methods
No methods found in currently loaded packages.
### equation

_Model equations_

**Description**

Display the mathematical representation of a fitted model.

**Usage**

`equation(object, ...)`

**Arguments**

- `object`: A fitted model object.
- `...`: Other arguments passed to methods

**Value**

Markup output suitable for rendering the equation.

**Methods**

No methods found in currently loaded packages.

### estfun

_Extracting the estimating functions of a fitted model._

**Description**

Extracting the estimating functions of a fitted model.

**Usage**

`estfun(x, ...)`

**Arguments**

- `x`: A fitted model object.
- `...`: Other arguments passed to methods

**Methods**

No methods found in currently loaded packages.
Evaluate an object.

**Description**
Evaluate an object.

**Usage**
evaluate(x, ...)

**Arguments**
x An object. See the individual method for specifics.
...
other arguments passed to methods

**Methods**
No methods found in currently loaded packages.

Explain details of an object

**Description**
Explain details of an object

**Usage**
explain(x, ...)

**Arguments**
x An object. See the individual method for specifics.
...
other arguments passed to methods

**Methods**
No methods found in currently loaded packages.
fit

Estimate model parameters.

Description

Estimates parameters for a given model from a set of data.

Usage

fit(object, ...)

Arguments

object An object. See the individual method for specifics.
...

Methods

No methods found in currently loaded packages.

fit_xy

Estimate model parameters.

Description

Estimates parameters for a given model from a set of data in the form of a set of predictors (x) and outcome(s) (y).

Usage

fit_xy(object, ...)

Arguments

object An object. See the individual method for specifics.
...

Methods

No methods found in currently loaded packages.
**generate**

*Generate values based on inputs*

**Description**

Generate values based on inputs

**Usage**

```r
generate(x, ...)
```

**Arguments**

- `x`  
  An object.

- `...`  
  Other arguments passed to methods

**Methods**

No methods found in currently loaded packages.

---

**glance**

*Glance at an object*

**Description**

`#' Construct a single row summary "glance" of a model, fit, or other object`

**Usage**

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

**Arguments**

- `x`  
  model or other R object to convert to single-row data frame

- `...`  
  other arguments passed to methods

**Details**

- glance methods always return either a one-row data frame (except on NULL, which returns an empty data frame)

**Methods**

No methods found in currently loaded packages.
hypothesize

Construct hypotheses.

Description

Construct hypotheses.

Usage

hypothesize(x, ...)

Arguments

x An object.
...
Other arguments passed to methods

Methods

No methods found in currently loaded packages.

interpolate

Interpolate missing values

Description

Interpolates missing values provided in the training dataset using the fitted model.

Usage

interpolate(object, ...)

Arguments

object A fitted model object
...
Other arguments passed to methods

Value

A dataset (tibble::tibble) or similar) of the same structure as the input dataset with missing values from the response variable replaced with interpolated values.

Methods

No methods found in currently loaded packages.
**learn**  
*Estimate model parameters.*

**Description**

Estimates parameters for a given model from a set of data.

**Usage**

`learn(x, ...)

**Arguments**

- `x`: An object. See the individual method for specifics.
- `...`: other arguments passed to methods

**Methods**

No methods found in currently loaded packages.

---

**prune**  
*Prune or reduce an object*

**Description**

Prune or reduce an object

**Usage**

`prune(tree, ...)

**Arguments**

- `tree`: A fitted model object.
- `...`: Other arguments passed to methods

**Methods**

No methods found in currently loaded packages.
### refit

**Description**
Refitting models

**Usage**

```r
refit(object, ...)  
```

**Arguments**

- `object`: A fitted model object.
- `...`: Other arguments passed to methods

**Methods**

No methods found in currently loaded packages.

### setops

**Description**

Union (`union()`), intersect (`intersect()`), difference (`setdiff()`), and equality (`setequal()`) for two vectors representing sets. Determine membership with `is.element()`.

**Usage**

```r
intersect(x, y, ...)  
union(x, y, ...)  
setdiff(x, y, ...)  
setequal(x, y, ...)  
is.element(el, set, ...)  
```

**Arguments**

- `x, y`: Vectors to combine.
- `...`: Other arguments passed on to methods.
- `el, set`: Element and set to compare.

**Details**

These functions override the set functions provided in base to make them generic so that packages can provide methods for different data types. The default methods call the base versions.
Value

For `union()`, `intersect()`, and `setdiff()`, a vector with all duplicate removed.
For `setequal()` and `is.element()`, a logical TRUE or FALSE.

Methods

`intersect()`: No methods found in currently loaded packages.

`union()`: No methods found in currently loaded packages.

`setdiff()`: No methods found in currently loaded packages.

`setequal()`: No methods found in currently loaded packages.

`is.element()`: No methods found in currently loaded packages.

Examples

```r
intersect(1:5, 4:8)
union(1:5, 4:8)
setdiff(1:5, 4:8)
setdiff(4:8, 1:5)
```

---

**specify**

*Specify variables or other quantities.*

Description

Specify variables or other quantities.

Usage

```r
specify(x, ...)
```

Arguments

- `x`  
  An object.

- `...`  
  Other arguments passed to methods

Methods

No methods found in currently loaded packages.
tidy  
*Turn an object into a tidy tibble*

Description

Turn an object into a tidy tibble

Usage

```r
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.

---

train  
*Estimate model parameters.*

Description

Estimates parameters for a given model from a set of data.

Usage

```r
train(x, ...)
```

Arguments

- `x`  
  An object. See the individual method for specifics.

- `...`  
  Other arguments passed to methods

Methods

No methods found in currently loaded packages.
varying_args

Find any arguments that are not fully specified.

Description
Find any arguments that are not fully specified.

Usage
`varying_args(object, ...)`

Arguments
- `object`: An object. See the individual method for specifics.
- `...`: Other arguments passed to methods

Methods
No methods found in currently loaded packages.

---

var_imp

Calculation of variable importance

Description
A generic method for calculating variable importance for model objects.

Usage
`var_imp(object, ...)`

Arguments
- `object`: A fitted model object.
- `...`: Other arguments passed to methods

Methods
No methods found in currently loaded packages.
visualize  

Visualize a data set or object.

Description

Visualize a data set or object.

Usage

visualize(x, ...)

Arguments

x  A data frame or other object.
...

Other arguments passed to methods

Methods

No methods found in currently loaded packages.
Index

as.difftime (coercion-time-difference), 4
as.factor (coercion-factor), 3
as.ordered (coercion-factor), 3
augment, 2
calculate, 2
coefficient-factor, 3
coefficient-time-difference, 4
compile, 5
components, 5
equation, 6
estfun, 6
evaluate, 7
explain, 7
fit, 8
fit_xy, 8
generate, 9
glance, 9
hypothesize, 10
interpolate, 10
intersect (setops), 12
is.element (setops), 12
learn, 11
prune, 11
refit, 12
setdiff (setops), 12
setequal (setops), 12
setops, 12
specify, 13
tibble::tibble(), 2, 5, 10, 14
tidy, 14
train, 14
union (setops), 12
var_imp, 15
varying_args, 15
visualize, 16