Package ‘modelsummary’

May 26, 2020

**Type** Package

**Title** Summary Tables for Statistical Models: Beautiful, Customizable, and Publication-Ready

**Description** Create beautiful and customizable summary tables for statistical models. 'modelsummary' leverages the power of the 'gt', 'kableExtra' and 'broom' packages. It supports dozens of model types, and can produce tables in HTML, LaTeX, RTF, Text/Markdown, JPG, PNG, and LaTeX formats. The tables can also be integrated in 'Rmarkdown', 'knitr', or 'Sweave' dynamic documents.

**Version** 0.3.0

**URL** https://vincentarelbundock.github.io/modelsummary

**BugReports** https://github.com/vincentarelbundock/modelsummary/issues

**Depends** R (>= 3.4.0)

**Imports** dplyr (>= 0.8.5), generics (>= 0.0.2), broom (>= 0.5.1), tidyr (>= 1.0.0), stringr (>= 1.3.0), knitr (>= 1.20), purrr (>= 0.2.1), checkmate (>= 1.8.5), magrittr (>= 1.5), tibble (>= 1.4.2), kableExtra (>= 1.1.0), gt (>= 0.2.0)

**Suggests** lmtest, MASS, sandwich, testthat, rmarkdown

**Enhances** mice, mitools, Amelia

**License** GPL-3

**Encoding** UTF-8

**LazyData** false

**RoxygenNote** 7.1.0

**NeedsCompilation** no

**Author** Vincent Arel-Bundock [aut, cre]

(https://orcid.org/0000-0003-2042-7063>)

**Maintainer** Vincent Arel-Bundock <vincent.arel-bundock@umontreal.ca>

**Repository** CRAN

**Date/Publication** 2020-05-26 17:50:02 UTC
### R topics documented:

- `clean_latex` ................................. 2
- `extract` .................................... 2
- `gof_map` .................................... 4
- `knit_latex` .................................. 5
- `modelsummary` ......................... 5
- `msummary` ................................. 8

---

#### `clean_latex`  
Deprecated function

**Description**

The `gt::as_latex` function does not (yet) produce compilable LaTeX code. This function used to clean up LaTeX output to allow compilation to PDF. This function is now deprecated since `modelsummary` currently supports `kableExtra`, which has a mature LaTeX rendering engine.

**Usage**

```r
clean_latex(...)
```

**Arguments**

- `...` catch everything

---

#### `extract`

*Extract and combine estimates and goodness-of-fit statistics from several statistical models.*

**Description**

Extract and combine estimates and goodness-of-fit statistics from several statistical models.

**Usage**

```r
eextract(
    models,
    statistic = "std.error",
    statistic_override = NULL,
    statistic_vertical = TRUE,
    conf_level = 0.95,
    coef_map = NULL,
    coef_omit = NULL,
    gof_map = modelsummary::gof_map,
```
gof_omit = NULL,
add_rows = NULL,
add_rows_location = NULL,
stars = FALSE,
fmt = "%.3f",
...
)

Arguments

models a single model object or a (potentially named) list of models to summarize
statistic string name of the statistic to include in parentheses
  • Typical values: "conf.int", "std.error", "statistic", "p.value"
  • Alternative values: any column name produced by ‘broom::tidy(model)’
statistic_override manually override the uncertainty estimates. This argument accepts three types of input:
  • a function or list of functions of length(models) which produce variance-covariance matrices with row and column names equal to the names of your coefficient estimates. For example, 'R' supplies the 'vcov' function, and the 'sandwich' package supplies 'vcovHC', 'vcovHAC', etc.
  • a list of length(models) variance-covariance matrices with row and column names equal to the names of your coefficient estimates.
  • a list of length(models) vectors with names equal to the names of your coefficient estimates. Numeric vectors are formatted according to 'fmt' and placed in brackets, character vectors printed as given.
statistic_vertical TRUE if statistics should be printed below estimates. FALSE if statistics should be printed beside estimates.
conf_level confidence level to use for confidence intervals
coef_map named character vector. Names refer to the original variable names. Values refer to the variable names that will appear in the table. Coefficients which are omitted from this vector will be omitted from the table. The table will be ordered in the same order as this vector.
coef_omit string regular expression. Omits all matching coefficients from the table (using ‘stringr::str_detect’).
gof_map data.frame with four columns: 'raw', 'clean', 'fmt', and 'omit'. See 'modelsummary::gof_map'
gof_omit string regular expression. Omits all matching gof statistics from the table (using ‘stringr::str_detect’).
add_rows list of character vectors, each of length equal to the number of models + 1.
add_rows_location integer or NULL, custom rows will be added to the bottom of the table if this parameter is NULL, or after the position set by this integer.
stars to indicate statistical significance
• FALSE (default): no significance stars.
• TRUE: *=.1, **=.05, ***=.01
• Named numeric vector for custom stars such as 'c('**' = .1, '+' = .05')'

fmt

string which specifies how numeric values will be rounded. This string is passed to the 'sprintf' function. '.3f' will keep 3 digits after the decimal point with trailing zero. '.5f' will keep 5 digits. '.3e' will use exponential notation. See '?sprintf' for more options.

... all other arguments are passed to the 'tidy' method used to extract estimates from the model. For example, this allows users to set 'exponentiate=TRUE' to exponentiate logistic regression coefficients.

Value
tibble

Examples

library(modelsummary)
data(trees)
models <- list()
models[['Bivariate']] <- lm(Girth ~ Height, data = trees)
models[['Multivariate']] <- lm(Girth ~ Height + Volume, data = trees)
extract(models)

---

gof_map

Data.frame used to clean up and format goodness-of-fit statistics

Description

By default, this data frame is passed to the 'gof_map' argument of the 'msummary' or 'modelsummary' functions. Users can modify this data frame to customize the list of statistics to display and their format. See example below.

Usage
gof_map

Format
data.frame with 4 columns of character data: raw, clean, fmt, omit
Examples

```r
library(modelsummary)
mod <- lm(wt ~ drat, data = mtcars)
gm <- modelsummary::gof_map
gm$omit[gm$raw == 'deviance'] <- FALSE
gm$fmt[gm$raw == 'r.squared'] <- "%5f"
msummary(mod, gof_map = gm)
```

Description

The `gt::as_latex` function does not (yet) produce compilable LaTeX code. This function used to clean up LaTeX output to allow compilation to PDF. This function is now deprecated since `modelsummary` currently supports `kableExtra`, which has a mature LaTeX rendering engine.

Usage

```r
knit_latex(...)```

Arguments

`...` catch everything

**Deprecated function**

**Beautiful, customizable summaries of statistical models**

Description

Beautiful, customizable summaries of statistical models

Usage

```r
modelsummary(
    models,
    output = "default",
    fmt = "%3f",
    statistic = "std.error",
    statistic_override = NULL,
    statistic_vertical = TRUE,
    conf_level = 0.95,
    stars = FALSE,
    coef_map = NULL,
    coef_omit = NULL,
    gof_map = modelsummary::gof_map,
```
gof_omit = NULL,
add_rows = NULL,
add_rows_location = NULL,
title = NULL,
notes = NULL,
filename = NULL,
subtitle = NULL,
... 
)

Arguments

models
a single model object or a (potentially named) list of models to summarize

output
filename or object type (string)

• Supported filename extensions: .html, .tex, .md, .txt, .png, .jpg.
• Supported object types: "gt", "html", "markdown", "latex". "gt" objects are created by the 'gt' package; other object types are created by the 'kableExtra' package.
• When a file name is supplied to the 'output' argument, the table is written immediately to file. If you want to customize your table by post-processing it with functions provided by the 'gt' or 'kableExtra' packages, you need to choose a different output format (e.g., "gt", "latex", "html", "markdown"), and you need to save the table after post-processing using the 'gt::gtsave', 'kable::save_kable', or 'cat' functions.

fmt
string which specifies how numeric values will be rounded. This string is passed to the 'sprintf' function. '%.3f' will keep 3 digits after the decimal point with trailing zero. '%.5f' will keep 5 digits. '%.3e' will use exponential notation. See '?sprintf' for more options.

statistic
string name of the statistic to include in parentheses

• Typical values: "conf.int", "std.error", "statistic", "p.value"
• Alternative values: any column name produced by 'broom::tidy(model)'

statistic_override
manually override the uncertainty estimates. This argument accepts three types of input:

• a function or list of functions of length(models) which produce variance-covariance matrices with row and column names equal to the names of your coefficient estimates. For example, 'R' supplies the 'vcov' function, and the 'sandwich' package supplies 'vcovHC', 'vcovHAC', etc.
• a list of length(models) variance-covariance matrices with row and column names equal to the names of your coefficient estimates.
• a list of length(models) vectors with names equal to the names of your coefficient estimates. Numeric vectors are formatted according to 'fmt' and placed in brackets, character vectors printed as given.

statistic_vertical
TRUE if statistics should be printed below estimates. FALSE if statistics should be printed beside estimates.
conf_level  confidence level to use for confidence intervals
stars        to indicate statistical significance
             • FALSE (default): no significance stars.
             • TRUE: *=.1, **=.05, ***=.01
             • Named numeric vector for custom stars such as ‘c("*" = .1, "+" = .05)’
coef_map     named character vector. Names refer to the original variable names. Values
             refer to the variable names that will appear in the table. Coefficients which are
             omitted from this vector will be omitted from the table. The table will be ordered
             in the same order as this vector.
coef_omit    string regular expression. Omits all matching coefficients from the table (using
             ‘stringr::str_detect’).
gof_map      data.frame with four columns: ‘raw’, ‘clean’, ‘fmt’, and ‘omit’. See ‘model-
             summary::gof_map’
gof_omit     string regular expression. Omits all matching gof statistics from the table (using
             ‘stringr::str_detect’).
add_rows     list of character vectors, each of length equal to the number of models + 1.
add_rows_location integer or NULL. custom rows will be added to the bottom of the table if this
             parameter is NULL, or after the position set by this integer.
title        string
notes        list of notes to append to the bottom of the table.
filename     This argument was deprecated in favor of the ‘output’ argument.
subtitle     This argument is deprecated. Use ‘title’ or the ‘tab_header’ function from the
             ‘gt’ package.
             ... all other arguments are passed to the ‘tidy’ method used to extract estimates
             from the model. For example, this allows users to set ‘exponentiate=TRUE’ to
             exponentiate logistic regression coefficients.

Value

a ‘gt’ table object.

Examples

# load data and estimate models
data(trees)
models <- list()
models[['Bivariate']] <- lm(Girth ~ Height, data = trees)
models[['Multivariate']] <- lm(Girth ~ Height + Volume, data = trees)

# simple table
msummary(models)

# confidence intervals, p values, or t-stats instead of standard errors
msummary(models, statistic = 'conf.int', conf_level = 0.99)
msummary(models, statistic = 'p.value', conf_level = 0.99)
msummary(models, statistic = 'statistic', conf_level = 0.99)

# rename and re-order coefficients
msummary(models, coef_map = c('Volume' = 'Large', 'Height' = 'Tall'))

# titles
msummary(models, title = 'This is the title')

# title with italicized text
msummary(models, title = gt::md('This is *the* title'))

# notes at the bottom of the table (here, the second note includes markdown bold characters)
msummary(models, notes = list('A first note', gt::md('A **bold** note')))

# modify list of GOF statistics and their format using the built-in
# 'gof_map' data frame as a starting point

gof_custom <- modelsummary::gof_map
gof_custom$omit[gof_custom$raw == 'deviance'] <- FALSE

gof_custom$fmt[gof_custom$raw == 'r.squared'] <- '%.5f'
msummary(models, gof_map = gof_custom)

---

msummary

Beautiful, customizable summaries of statistical models

Description

'msummary()' is a shortcut to 'modelsummary()'

Usage

msummary(
  models, 
  output = "default", 
  fmt = "%.3f", 
  statistic = "std.error", 
  statistic_override = NULL, 
  statistic_vertical = TRUE, 
  conf_level = 0.95, 
  stars = FALSE, 
  coef_map = NULL, 
  coef_omit = NULL, 
  gof_map = modelsummary::gof_map, 
  gof_omit = NULL, 
  add_rows = NULL, 
  add_rows_location = NULL,
)
Arguments

- **models**: a single model object or a (potentially named) list of models to summarize
- **output**: filename or object type (string)
  - Supported filename extensions: .html, .tex, .md, .txt, .png, .jpg.
  - Supported object types: "gt", "html", "markdown", "latex". "gt" objects are created by the 'gt' package; other object types are created by the 'kableExtra' package.
  - When a filename is supplied to the 'output' argument, the table is written immediately to file. If you want to customize your table by post-processing it with functions provided by the 'gt' or 'kableExtra' packages, you need to choose a different output format (e.g., "gt", "latex", "html", "markdown"), and you need to save the table after post-processing using the 'gt::gtsave', 'kable::save_kable', or 'cat' functions.
- **fmt**: string which specifies how numeric values will be rounded. This string is passed to the 'sprintf' function. '%.3f' will keep 3 digits after the decimal point with trailing zero. '%.5f' will keep 5 digits. '%.3e' will use exponential notation. See '?sprintf' for more options.
- **statistic**: string name of the statistic to include in parentheses
  - Typical values: "conf.int", "std.error", "statistic", "p.value"
  - Alternative values: any column name produced by 'broom::tidy(model)'
- **statistic_override**: manually override the uncertainty estimates. This argument accepts three types of input:
  - a function or list of functions of length(models) which produce variance-covariance matrices with row and column names equal to the names of your coefficient estimates. For example, 'R' supplies the 'vcov' function, and the 'sandwich' package supplies 'vcovHC', 'vcovHAC', etc.
  - a list of length(models) variance-covariance matrices with row and column names equal to the names of your coefficient estimates.
  - a list of length(models) vectors with names equal to the names of your coefficient estimates. Numeric vectors are formatted according to 'fmt' and placed in brackets, character vectors printed as given.
- **statistic_vertical**: TRUE if statistics should be printed below estimates. FALSE if statistics should be printed beside estimates.
- **conf_level**: confidence level to use for confidence intervals
- **stars**: to indicate statistical significance
msummary

- FALSE (default): no significance stars.
- TRUE: * = .1, ** = .05, *** = .01
- Named numeric vector for custom stars such as 'c("*" = .1, "+" = .05)'

**coef_map**

Named character vector. Names refer to the original variable names. Values refer to the variable names that will appear in the table. Coefficients which are omitted from this vector will be omitted from the table. The table will be ordered in the same order as this vector.

**coef_omit**

String regular expression. Omits all matching coefficients from the table (using `stringr::str_detect`).

**gof_map**

Data frame with four columns: `raw`, `clean`, `fmt`, and `omit`. See `modelsummary::gof_map`.

**gof_omit**

String regular expression. Omits all matching gof statistics from the table (using `stringr::str_detect`).

**add_rows**

List of character vectors, each of length equal to the number of models + 1.

**add_rows_location**

Integer or NULL. Custom rows will be added to the bottom of the table if this parameter is NULL, or after the position set by this integer.

**title**

String

**notes**

List of notes to append to the bottom of the table.

**filename**

This argument was deprecated in favor of the 'output' argument.

**subtitle**

This argument is deprecated. Use 'title' or the 'tab_header' function from the 'gt' package.

... all other arguments are passed to the 'tidy' method used to extract estimates from the model. For example, this allows users to set 'exponentiate=TRUE' to exponentiate logistic regression coefficients.

**Value**

A 'gt' table object.

**Examples**

```r
# load data and estimate models
data(trees)
models <- list()
models[['Bivariate']] <- lm(Girth ~ Height, data = trees)
models[['Multivariate']] <- lm(Girth ~ Height + Volume, data = trees)

# simple table
msummary(models)

# confidence intervals, p values, or t-stats instead of standard errors
msummary(models, statistic = 'conf.int', conf_level = 0.99)
msummary(models, statistic = 'p.value', conf_level = 0.99)
msummary(models, statistic = 'statistic', conf_level = 0.99)
```
# rename and re-order coefficients
msummary(models, coef_map = c('Volume' = 'Large', 'Height' = 'Tall'))

# titles
msummary(models, title = 'This is the title')

# title with italicized text
msummary(models, title = gt::md('This is *the* title'))

# notes at the bottom of the table (here, the second note includes markdown bold characters)
msummary(models, notes = list('A first note', gt::md('A **bold** note')))

# modify list of GOF statistics and their format using the built-in
gof_custom <- modelsummary::gof_map
gof_custom$omit[gof_custom$raw == 'deviance'] <- FALSE
gof_custom$fnt[gof_custom$raw == 'r.squared'] <- '%.5f'
msummary(models, gof_map = gof_custom)
Index

* Topic **datasets**
  - *gof_map*, 4

  *clean_latex*, 2
  *extract*, 2
  *gof_map*, 4
  *knit_latex*, 5
  *modelsummary*, 5
  *msummary*, 8