

Package ‘likert’

May 8, 2026

Type Package

Title Analysis and Visualization Likert Items

Version 1.3.5.1

Date 2016-12-29

URL <https://github.com/jbryer/likert>

BugReports <https://github.com/jbryer/likert/issues>

Description An approach to analyzing Likert response items, with an emphasis on visualizations. The stacked bar plot is the preferred method for presenting Likert results. Tabular results are also implemented along with density plots to assist researchers in determining whether Likert responses can be used quantitatively instead of qualitatively. See the `likert()`, `summary.likert()`, and `plot.likert()` functions to get started.

License GPL

LazyLoad yes

VignetteBuilder utils

Depends R (>= 3.0),ggplot2,xtable

Imports psych,reshape2,gridExtra,grid,plyr

Suggests devtools,shiny

RoxygenNote 7.3.2

NeedsCompilation no

Repository CRAN

Date/Publication 2025-08-02 08:02:55 UTC

Author Jason Bryer [aut, cre],
Kimberly Speerschneider [aut]

Maintainer Jason Bryer <jason@bryer.org>

Contents

likert-package	2
abs_formatter	3

align.plots	3
gap	4
label_wrap_mod	4
likert	4
likert.bar.plot	6
likert.density.plot	7
likert.heat.plot	7
likert.histogram.plot	8
likert.matrix.plot	9
likert.options	10
mass	11
MathAnxiety	12
MathAnxietyGender	12
pisaitems	13
plot.likert	13
plot.likert.gap	14
print.likert	16
print.likert.bar.plot	16
print.likert.gap	17
print.likert.heat.plot	17
print.xlikert	18
recode	18
reverse.levels	19
sasr	19
shinyLikert	20
summary.likert	20
summary.likert.gap	21
xtable.likert	21

Index **23**

likert-package	<i>Likert Analysis and Visualization</i>
----------------	--

Description

Likert Analysis and Visualization

Author(s)

<jason@bryer.org>

See Also

Useful links:

- <https://github.com/jbryer/likert>
- Report bugs at <https://github.com/jbryer/likert/issues>

abs_formatter	<i>Absolute value formatter for continuous_scale</i>
---------------	--

Description

This will print the absolute value for labeling on axis. Usefull for stacked bar plots where negative values are not negative percentages but represent negative groups.

Usage

```
abs_formatter(x)
```

Arguments

x	value to be reformatted.
---	--------------------------

Value

the absolute value of x.

align.plots	<i>Adapted from ggExtra package which is no longer available. This is related to an experimental mlpsa plot that will combine the circular plot along with the two individual distributions.</i>
-------------	--

Description

Adapted from ggExtra package which is no longer available. This is related to an experimental mlpsa plot that will combine the circular plot along with the two individual distributions.

Usage

```
## S3 method for class 'plots'
align(gl, ...)
```

Arguments

gl	grid.layout
...	graphic elements to combine.

References

http://groups.google.com/group/ggplot2/browse_thread/thread/1b859d6b4b441c90 <http://ggextra.googlecode.com/svn/trunk>

gap	<i>Fictitious dataset with importance and satisfaction results across five different offices.</i>
-----	---

Description

This data set is used in the GapAnalysis demo and is used to demonstrate how the likert package handles a gap analysis.

Format

a data frame with 68 observations of 11 variables.

label_wrap_mod	<i>Wrap label text.</i>
----------------	-------------------------

Description

Wrap label text.

Usage

```
label_wrap_mod(value, width = 25)
```

Arguments

value	vector (converted using as.character) to be wrapped.
width	the maximum width of each line in characters. Adapted from https://github.com/hadley/ggplot2/wiki/labeller

likert	<i>Analyze Likert type items.</i>
--------	-----------------------------------

Description

This function will provide various statistics about a set of likert items. The resulting object will have the following items:

Usage

```
likert(  
  items,  
  summary,  
  grouping = NULL,  
  factors = NULL,  
  importance,  
  nlevels = length(levels(items[, 1]))  
)
```

Arguments

<code>items</code>	data frame containing the likert based items. The variables in the data frame should be factors.
<code>summary</code>	a pre-summarized data frame. The first column must be the items and the remaining columns are the levels (e.g. strongly disagree, disagree, etc).
<code>grouping</code>	(optional) should the results be summarized by the given grouping variable.
<code>factors</code>	a vector with <code>length(factors) == ncol(items)</code> defining which factor each column belongs to. The values correspond to the factor label.
<code>importance</code>	a data frame of the same dimensions as <code>items</code> containing an importance rating for each item. The order of columns should match and the names from <code>items</code> will be used.
<code>nlevels</code>	number of possible levels. Only necessary if there are missing levels.

Details

- `results` - this data frame will contain a column 'Item', 'Group' (if a grouping variable was specified, and a column for each level of the items (e.g. agree, disagree, etc.). The value within each cell corresponds to the percentage of responses for that level and group.
- `items` - a copy of the original items data frame.
- `grouping` - a copy of the original grouping vector.
- `nlevels` - the number of levels used in the calculations.

Value

a likert class with the following elements: `results`, `items`, `grouping`, `nlevels`, and `summary`.

See Also

`plot.likert`
`summary.likert`

Examples

```

data(pisaitems)
items29 <- pisaitems[,substr(names(pisaitems), 1,5) == 'ST25Q']
names(items29) <- c("Magazines", "Comic books", "Fiction",
                  "Non-fiction books", "Newspapers")
l29 <- likert(items29)
summary(l29)
plot(l29)

```

likert.bar.plot	<i>Bar Plot for Likert Items.</i>
-----------------	-----------------------------------

Description

Bar plot for the results of [likert](#).

Usage

```
likert.bar.plot(l, group.order, center = (l$levels - 1)/2 + 1, ...)
```

Arguments

l	results of likert .
group.order	the order in which groups (for grouped items) or items (for non-grouped items) should be plotted.
center	specifies which level should be treated as the center. For example, center = 3 would use the third level as the center whereas center = 3.5 would indicate no specific level is the center but ≤ 3 are low levels and ≥ 4 are high levels (i.e. used for forced choice items or those without a neutral option). This also influences where the color breaks from low to high.
...	passed to likert.options
likert	object of type likert.

See Also

[plot.likert](#)
[likert.heat.plot](#)
[likert.bar.plot](#)
[likert.density.plot](#)

likert.density.plot *Creates a density plot for likert items.*

Description

This function will create a visualization that treats the likert items as a continuous variable.

Usage

```
likert.density.plot(likert, facet = TRUE, bw = 0.5, legend, ...)
```

Arguments

likert	object of type likert.
facet	for non-grouped items, should each density distribution be plotted in a separate facet.
bw	the smoothing bandwidth. This is often set to the standard deviation but this is often inadequate for Likert type items. The value of 0.5 is used since the difference between any two adjacent levels is one.
legend	title for the legend.
...	parameters passed to density .

See Also

plot.likert

likert.heat.plot *Internal method.*

Description

Internal method.

Usage

```
likert.heat.plot(  
  likert,  
  low.color = "white",  
  high.color = "blue",  
  text.color = "black",  
  text.size = 4,  
  wrap = 50,  
  ...  
)
```

Arguments

likert	object of type likert.
low.color	color for low values.
high.color	color for high values.
text.color	color of text attributes.
text.size	size of text attributes.
wrap	width to wrap label text for non-grouped likert objects.
...	currently unused.

See Also

plot.likert
likert.bar.plot

likert.histogram.plot *Histogram of number of responses.*

Description

Plots a histogram of the number of responses for each item and group (if specified). Negative values (in maroon by default) indicate the number of missing values for that item and group.

Usage

```
likert.histogram.plot(  
  l,  
  xlab = "n",  
  plot.missing = TRUE,  
  bar.color = "grey70",  
  missing.bar.color = "maroon",  
  label.completed = "Completed",  
  label.missing = "Missing",  
  legend.position = "bottom",  
  wrap = ifelse(is.null(l$grouping), 50, 100),  
  order,  
  group.order,  
  panel.arrange = "v",  
  panel.strip.color = "#F0F0F0",  
  text.size = 2.5,  
  ...  
)
```

Arguments

l	results of <code>likert</code> .
xlab	label used for the x-axis.
plot.missing	if TRUE, missing values will be plotted to the left of the x-axis.
bar.color	the bar color.
missing.bar.color	the color of the bar for missing values.
label.completed	the label to use in the legend representing the count of complete values.
label.missing	the label to use in the legend representing the count of missing values.
legend.position	location of the legend.
wrap	number of characters before warping the text in the panel strips.
order	the order of the items.
group.order	the order in which groups (for grouped items) or items (for non-grouped items) should be plotted.
panel.arrange	v for vertical or h for horizontal.
panel.strip.color	the color for panels.
text.size	text size.
...	other ggplot2 parameters.

likert.matrix.plot	<i>Matrix plot (experimental)</i>
--------------------	-----------------------------------

Description

Matrix plot (experimental)

Usage

```
likert.matrix.plot(likert, nSample = nrow(likert$items), ...)
```

Arguments

likert	results of <code>likert</code> .
nSample	random sample of all rows. This function may take a while to run with large datasets (including the <code>pisaitems</code> data). Plotting a random subsample allows for quicker development.
...	parameters passed to <code>pairs.ordered.categorical</code> .

likert.options *Builds an object with options for plotting [likert](#) results.*

Description

Builds an object with options for plotting [likert](#) results.

Usage

```
likert.options(  
  low.color = "#D8B365",  
  high.color = "#5AB4AC",  
  neutral.color = "grey90",  
  neutral.color.ramp = "white",  
  colors = NULL,  
  plot.percent.low = TRUE,  
  plot.percent.high = TRUE,  
  plot.percent.neutral = TRUE,  
  plot.percentages = FALSE,  
  text.size = 3,  
  text.color = "black",  
  centered = TRUE,  
  include.center = TRUE,  
  ordered = TRUE,  
  wrap = 50,  
  wrap.grouping = 50,  
  legend = "Response",  
  legend.position = "bottom",  
  panel.arrange = "v",  
  panel.strip.color = "#F0F0F0",  
  ...  
)
```

Arguments

<code>low.color</code>	color for low values.
<code>high.color</code>	color for high values.
<code>neutral.color</code>	color for middle values (if odd number of levels).
<code>neutral.color.ramp</code>	second color used when calling colorRamp with <code>low.color</code> and <code>high.color</code> to define the color palettes.
<code>colors</code>	vector specifying the colors to use. This must be equal to the number of likert levels.
<code>plot.percent.low</code>	whether to plot low percentages.

<code>plot.percent.high</code>	whether to plot high percentages.
<code>plot.percent.neutral</code>	whether to plot neutral percentages.
<code>plot.percents</code>	whether to label each category/bar.
<code>text.size</code>	size of text attributes.
<code>text.color</code>	color of text attributes.
<code>centered</code>	if true, the bar plot will be centered around zero such that the lower half of levels will be negative.
<code>include.center</code>	if TRUE, include the center level in the plot otherwise the center will be excluded.
<code>ordered</code>	reorder items from high to low.
<code>wrap</code>	width to wrap label text for item labels
<code>wrap.grouping</code>	width to wrap label text for group labels.
<code>legend</code>	title for the legend.
<code>legend.position</code>	the position for the legend ("left", "right", "bottom", "top", or two-element numeric vector).
<code>panel.arrange</code>	how panels for grouped likert items should be arrange. Possible values are v (vertical, the default), h (horizontal), and NULL (auto fill horizontal and vertical)
<code>panel.strip.color</code>	the background color for panel labels.
<code>...</code>	included for future expansion.

 mass

Results from an administration of the Math Anxiety Scale Survey.

Description

A data frame of results of the Math Anxiety Scale Survey administered to 20 students in a statistics course. This data frame contains the original data and can be used to verify the pre-summarized procedures.

Format

data frame with 14 rows and 6 columns.

References

Bai, H., Wang, L., Pan, W., & Frey, M. (2009). Measuring mathematics anxiety: Psychometric analysis of a bidimensional affective scale. *Journal of Instructional Psychology*, 36 (3), 185- 193.

MathAnxiety	<i>Pre-summarized results from an administration of the Math Anxiety Scale Survey.</i>
-------------	--

Description

A data frame of presummarized results of the Math Anxiety Scale Survey administered to 20 students in a statistics course.

Format

data frame with 14 rows and 6 columns.

References

Bai, H., Wang, L., Pan, W., & Frey, M. (2009). Measuring mathematics anxiety: Psychometric analysis of a bidimensional affective scale. *Journal of Instructional Psychology*, 36 (3), 185- 193.

MathAnxietyGender	<i>Pre-summarized results from an administration of the Math Anxiety Scale Survey grouped by gender.</i>
-------------------	--

Description

A data frame of presummarized results of the Math Anxiety Scale Survey administered to 20 students in a statistics course grouped by gender.

Format

data frame with 28 rows and 7 columns.

References

Bai, H., Wang, L., Pan, W., & Frey, M. (2009). Measuring mathematics anxiety: Psychometric analysis of a bidimensional affective scale. *Journal of Instructional Psychology*, 36 (3), 185- 193.

pisaitems

*Programme of International Student Assessment***Description**

North American (i.e. Canada, Mexico, and United States) results from the 2009 Programme of International Student Assessment (PISA) as provided by the Organization for Economic Co-operation and Development (OECD). See <http://www.pisa.oecd.org/> for more information including the code book.

Format

a data frame 66,690 observations of 81 variables from North America.

Source

Organization for Economic Co-operation and Development

plot.likert

*Plots a set of likert items.***Description**

This is an implementation of the S3 plot generic function. Based upon the type parameter this function will call either `likert.bar.plot`, `likert.heat.plot`, or `likert.density.plot`. See the help pages for those functions for all the available parameters to customize the aesthetics of the figure. Although those functions can be plotted directly, we recommend call the generic plot function.

Usage

```
## S3 method for class 'likert'
plot(
  x,
  type = c("bar", "heat", "density"),
  include.histogram = FALSE,
  panel.widths = c(3, 1),
  panel.arrange = "v",
  panel.strip.color = "#F0F0F0",
  legend.position = "bottom",
  group.order,
  panel.background = element_rect(size = 1, color = "grey70", fill = NA),
  ...
)
```

Arguments

x	the likert items to plot
type	the type of plot to create. Current values are bar and heat.
include.histogram	if TRUE, a histogram of count of responses is also plotted.
panel.widths	if include.histogram=TRUE, this vector of length two specifies the ratio of the left and right panels.
panel.arrange	how panels for grouped likert items should be arrange. Possible values are v (vertical, the default), h (horizontal), and NULL (auto fill horizontal and vertical)
panel.strip.color	the background color for panel labels.
legend.position	the position for the legend ("left", "right", "bottom", "top", or two-element numeric vector).
group.order	the order in which groups (for grouped items) or items (for non-grouped items) should be plotted.
panel.background	define background of the plot. See theme .
...	other parameters passed passed to likert.bar.plot or likert.heat.plot .

See Also

[likert.bar.plot](#)
[likert.heat.plot](#)
[likert.density.plot](#)
[likert.histogram.plot](#)

plot.likert.gap *Plots a set of likert items.*

Description

This is an implementation of the S3 plot generic function. Based upon the type parameter this function will call either [likert.bar.plot](#), [likert.heat.plot](#), or [likert.density.plot](#). See the help pages for those functions for all the available parameters to customize the aesthetics of the figure. Although those functions can be plotted directly, we recommend call the generic plot function.

Usage

```
## S3 method for class 'likert.gap'
plot(
  x,
  type = c("bar", "density"),
  include.histogram = FALSE,
  panel.widths = c(3, 1),
  panel.arrange = "v",
  panel.strip.color = "#F0F0F0",
  legend.position = "bottom",
  panel.background = element_rect(size = 1, color = "grey70", fill = NA),
  satisfaction.label = "Satisfaction",
  importance.label = "Importance",
  legend,
  ...
)
```

Arguments

<code>x</code>	the likert items to plot
<code>type</code>	the type of plot to create. Current values are bar and heat.
<code>include.histogram</code>	if TRUE, a histogram of count of responses is also plotted.
<code>panel.widths</code>	if <code>include.histogram=TRUE</code> , this vector of length two specifies the ratio of the left and right panels.
<code>panel.arrange</code>	how panels for grouped likert items should be arrange. Possible values are v (vertical, the default), h (horizontal), and NULL (auto fill horizontal and vertical)
<code>panel.strip.color</code>	the background color for panel labels.
<code>legend.position</code>	the position for the legend ("left", "right", "bottom", "top", or two-element numeric vector).
<code>panel.background</code>	define background of the plot. See theme .
<code>satisfaction.label</code>	label used for satisfaction items.
<code>importance.label</code>	label used for importance items.
<code>legend</code>	title for the legend.
<code>...</code>	other parameters passed passed to likert.bar.plot or likert.heat.plot .

See Also

[likert.bar.plot](#)

[likert.heat.plot](#)

`print.likert.gap` *Prints results table.*

Description

Prints results table.

Usage

```
## S3 method for class 'likert.gap'  
print(x, ...)
```

Arguments

`x` the likert class to print.
`...` parameters passed to `print.data.frame`.

`print.likert.heat.plot`
Print method for `likert.heat.plot`.

Description

Print method for `likert.heat.plot`.

Usage

```
## S3 method for class 'likert.heat.plot'  
print(p, ...)
```

Arguments

`p` a plot from `likert.heat.plot`.
`...` other parameters passed to `ggplot2`.

<code>print.xlikert</code>	<i>Prints the results of <code>xtable.likert</code>.</i>
----------------------------	--

Description

Print method for `xtable.likert`.

Usage

```
## S3 method for class 'xlikert'
print(x, tabular.environment = "longtable", floating = FALSE, ...)
```

Arguments

<code>x</code>	results of <code>xtable.likert</code> .
<code>tabular.environment</code>	see <code>print.xtable</code> .
<code>floating</code>	see <code>print.xtable</code> .
<code>...</code>	other parameters passed to <code>print.xtable</code>

<code>recode</code>	<i>Recode a vector.</i>
---------------------	-------------------------

Description

This utility function will recode values from an original `character` or `factor` vector with new values.

Usage

```
recode(x, from, to, to.class = NULL)
```

Arguments

<code>x</code>	the vector whose values will be recoded.
<code>from</code>	the old values in <code>x</code> to be recoded.
<code>to</code>	the new values.
<code>to.class</code>	an 'as.' function representing the desired vector type (i.e. <code>as.character</code> , <code>as.numeric</code> , <code>as.logical</code> , <code>as.numeric</code>).

Value

a vector with same length of `x` with recoded values.

Examples

```
test <- letters[sample(5, 10, replace=TRUE)]
recode(test, from=letters[1:5], to=paste('Letter', letters[1:5]))
```

reverse.levels	<i>Reverse the levels of a factor.</i>
----------------	--

Description

Reverse the levels of a factor.

Usage

```
reverse.levels(x)
```

Arguments

x a factor or a data.frame of factors whose levels will be reverse coded.

Examples

```
mylevels <- c('Strongly Disagree', 'Disagree', 'Neither', 'Agree', 'Strongly Agree')
test <- factor(sample(mylevels[1:5], 10, replace=TRUE))
cbind(test, as.integer(test), as.integer(reverse.levels(test)))
```

sasr	<i>Results from the Survey of Academic Self-Regulation (SASR).</i>
------	--

Description

The Survey of Academic Self-Regulation (SASR) is comprised of six factors: self-regulation, intrinsic motivation, extrinsic motivation, self-efficacy, metacognition, and personal relevance and control.

Format

a data frame with 860 observations of 63 variables.

References

Dugan, R., & Andrade, H. (2011). Exploring the construct validity of academic self-regulation using a new self-report questionnaire. *The International Journal of Educational and Psychological Assessment*, 7(1).

shinyLikert	<i>Shiny App for the likert package.</i>
-------------	--

Description

This will start a shiny app included with the package to show many of the features available in the likert package.

Usage

```
shinyLikert()
```

References

<http://rstudio.com/shiny>

summary.likert	<i>Prints summary table of a Likert analysis.</i>
----------------	---

Description

The summary function returns a data frame that provides additional information. It contains 'Item' and 'Group' columns similar to the results data frame as well as a column 'low' corresponding to the sum of levels below neutral, a column 'high' corresponding to the sum of levels above neutral, and columns 'mean' and 'sd' corresponding to the mean and standard deviation, respectively, of the results. The numeric values are determined by as.numeric which will use the values of the factors.

Usage

```
## S3 method for class 'likert'
summary(object, center = (object$nlevels - 1)/2 + 1, ordered = TRUE, ...)
```

Arguments

object	the likert class to summarize.
center	specifies which level should be treated as the center. For example, center = 3 would use the third level as the center whereas center = 3.5 would indicate no specific level is the center but ≤ 3 are low levels and ≥ 4 are high levels (i.e. used for forced choice items or those without a neutral option).
ordered	whether the results should be ordered. Currently unsupported for grouped analysis.
...	currently unused.

summary.likert.gap *Prints summary table of a Likert analysis.*

Description

The summary function returns a data frame that provides additional information. It contains 'Item' and 'Group' columns similar to the results data frame as well as a column 'low' corresponding to the sum of levels below neutral, a column 'high' corresponding to the sum of levels above neutral, and columns 'mean' and 'sd' corresponding to the mean and standard deviation, respectively, of the results. The numeric values are determined by `as.numeric` which will use the values of the factors.

Usage

```
## S3 method for class 'likert.gap'  
summary(object, ...)
```

Arguments

object the likert class to summarize.
... parameters passed to [summary.likert](#)

Value

a list with two data frames with summarized data for satisfaction and importance results separately.

xtable.likert *Prints a LaTeX table of the likert items.*

Description

Create a LaTeX or HTML table of the [likert](#) results.

Usage

```
## S3 method for class 'likert'  
xtable(  
  x,  
  caption = NULL,  
  label = NULL,  
  align = NULL,  
  digits = NULL,  
  display = NULL,  
  auto = FALSE,  
  include.n = TRUE,  
  include.mean = TRUE,
```

```

include.sd = TRUE,
include.low = TRUE,
include.neutral = (x$nlevels%%2 != 0),
include.high = TRUE,
include.levels = TRUE,
include.missing = TRUE,
center = (x$nlevels - 1)/2 + 1,
ordered = TRUE,
...
)

```

Arguments

x	likert class object.
caption	the table caption.
label	the table label.
align	column alignments.
digits	number of digits to use for numeric columns.
display	column formats.
auto	Logical, indicating whether to apply automatic format when no value is passed to align, digits, or display (see xtable for more information).
include.n	option to include n
include.mean	option to include mean
include.sd	option to include sd
include.low	option to include low
include.neutral	option to include neutral
include.high	option to include high
include.levels	option to include levels
include.missing	option to include missing levels.
center	specifies which level should be treated as the center. For example, center = 3 would use the third level as the center whereas center = 3.5 would indicate no specific level is the center but ≤ 3 are low levels and ≥ 4 are high levels (i.e. used for forced choice items or those without a neutral option). This also influences which levels are summarized in the low and high groups.
ordered	whether the results should be ordered. See summary.likert
...	other parameters passed to xtable .

See Also

[xtable](#), [print.xtable](#)

Index

- * **datasets**
 - gap, [4](#)
 - mass, [11](#)
 - MathAnxiety, [12](#)
 - MathAnxietyGender, [12](#)
 - pisaitems, [13](#)
 - sasr, [19](#)
- * **institutional**
 - likert-package, [2](#)
- * **likert**
 - likert-package, [2](#)
- * **package**
 - likert-package, [2](#)
- * **research**
 - likert-package, [2](#)
- abs_formatter, [3](#)
- align.plots, [3](#)
- as.character, [4](#)
- character, [18](#)
- colorRamp, [10](#)
- density, [7](#)
- factor, [18](#)
- gap, [4](#)
- label_wrap_mod, [4](#)
- likert, [4](#), [6](#), [9](#), [10](#), [21](#)
- likert-package, [2](#)
- likert.bar.plot, [6](#), [13–16](#)
- likert.density.plot, [7](#), [13](#), [14](#), [16](#)
- likert.heat.plot, [7](#), [13–15](#), [17](#)
- likert.histogram.plot, [8](#), [14](#), [16](#)
- likert.matrix.plot, [9](#)
- likert.options, [6](#), [10](#)
- mass, [11](#)
- MathAnxiety, [12](#)
- MathAnxietyGender, [12](#)
- pisaitems, [13](#)
- plot.likert, [13](#)
- plot.likert.gap, [14](#)
- print.data.frame, [16](#), [17](#)
- print.likert, [16](#)
- print.likert.bar.plot, [16](#)
- print.likert.gap, [17](#)
- print.likert.heat.plot, [17](#)
- print.xlikert, [18](#)
- print.xtable, [18](#), [22](#)
- recode, [18](#)
- reverse.levels, [19](#)
- sasr, [19](#)
- shinyLikert, [20](#)
- summary.likert, [20](#), [21](#), [22](#)
- summary.likert.gap, [21](#)
- theme, [14](#), [15](#)
- xtable, [22](#)
- xtable.likert, [18](#), [21](#)