

Package ‘visreg’

July 28, 2026

Title Visualization of Regression Models

Version 3.0.0

Description Provides a convenient interface for constructing plots to visualize the fit of regression models arising from a wide variety of models in R ('lm', 'glm', 'coxph', 'rlm', 'gam', 'locfit', 'lmer', 'randomForest', etc.)

License GPL-3

URL <https://pbreheny.github.io/visreg/>,
<https://github.com/pbreheny/visreg>

BugReports <https://github.com/pbreheny/visreg/issues>

Depends R (>= 4.1)

Imports rlang, ggplot2

Suggests glmmTMB, knitr, lattice, lme4, MASS, Matrix, quarto, rgl, survival, tinytest

Enhances nlme

VignetteBuilder quarto

SystemRequirements Quarto command line tool

Encoding UTF-8

Config/roxygen2/version 8.0.0

NeedsCompilation no

Author Patrick Breheny [aut, cre] (ORCID:
<<https://orcid.org/0000-0002-0650-1119>>),
Woodrow Burchett [ctb]

Maintainer Patrick Breheny <patrick-breheny@uiowa.edu>

Repository CRAN

Date/Publication 2026-07-28 17:10:02 UTC

Contents

collapse	2
persp.visreg2d	3
plot.visreg	4
plot.visreg2d	7
subset.visreg	8
visreg	9
visreg2d	13
visreg_list	15
Index	17

collapse	<i>Collapse a visreg_list into a single visreg object</i>
----------	---

Description

Takes a visreg_list, as produced by `visreg_list()` or `visreg()`, and collapses it down into a single visreg object. This is equivalent to calling `visreg_list()` with `collapse = TRUE` in the first place; it is provided separately so that a visreg_list can be collapsed after the fact, e.g. `visreg_list(v1, v2) |> collapse()`.

Usage

`collapse(x, labels)`

Arguments

- | | |
|---------------------|---|
| <code>x</code> | A visreg_list object. |
| <code>labels</code> | A character vector with length corresponding to the number of visreg objects in <code>x</code> that provides labels for the different objects in subsequent plots. If missing, the response variable names are used (or Y1, Y2, ... if these are not unique). |

Value

A visreg object.

See Also

`visreg_list()`, `visreg()`

Examples

```
fit <- lm(Ozone ~ Solar.R + Wind + Temp, data = airquality)
v1 <- visreg(fit, "Wind", plot = FALSE, alpha = 0.2)
v2 <- visreg(fit, "Wind", plot = FALSE, alpha = 0.01)
labs <- c("Confidence: 0.80", "Confidence: 0.99")
vv1 <- visreg_list(v1, v2, collapse = FALSE) |> collapse(labels = labs)
vv2 <- visreg_list(v1, v2, collapse = TRUE, labels = labs)
identical(vv1, vv2)
```

persp.visreg2d

Static 3D perspective plot for a visreg2d object

Description

persp() method for [visreg2d\(\)](#) objects: draws a static 3-dimensional perspective plot of the fitted surface, using base R's [graphics::persp\(\)](#).

Usage

```
## S3 method for class 'visreg2d'
persp(
  x,
  xlab = NULL,
  ylab = NULL,
  zlab = NULL,
  color = "#2fa4e7",
  whitespace = 0.2,
  ...
)

## S3 method for class 'visreg_list'
persp(x, ...)

persp3d.visreg_list(x, ...)
```

Arguments

x	A visreg2d() object.
xlab, ylab	Axis labels for predictors (default: variable name)
zlab	Axis label for outcome (default: variable name)
color	The color of the surface.
whitespace	When xvar or yvar is a factor, whitespace determines the amount of space in between the factors. Default is 0.2, meaning that 20 percent of the axis is whitespace.
...	Additional graphical parameters, passed to graphics::persp() .

Details

For an interactive version that can be rotated with the mouse, see `rgl::persp3d()` (requires the `rgl` package). For a 2-dimensional raster/contour plot, see `plot.visreg2d()`.

Value

The viewing transformation matrix, invisibly; see `graphics::persp()`.

References

Breheny P and Burchett W. (2017) Visualization of regression models using visreg. *R Journal*, **9**: 56-71. doi:10.32614/RJ2017046

See Also

`visreg2d()` for creating two-dimensional visreg objects, `plot.visreg2d()` for the 2-dimensional raster/contour plot, and the [surface plots vignette](#) for examples and details.

Examples

```
fit <- lm(
  Ozone ~ Solar.R + Wind + Temp +
    I(Wind^2) + I(Temp^2) + I(Wind * Temp) +
    I(Wind * Temp^2) + I(Temp * Wind^2) +
    I(Temp^2 * Wind^2),
  data = airquality
)

v <- visreg2d(fit, x = "Wind", y = "Temp", plot = FALSE)
persp(v)

airquality$Heat <- cut(airquality$Temp, 3, labels = c("Cool", "Mild", "Hot"))
fit2 <- lm(Ozone ~ Solar.R + Wind * Heat, data = airquality)
visreg2d(fit2, "Heat", "Wind", plot = FALSE) |> persp(theta = 230)
```

Description

Default plots contain a confidence band, prediction line, and partial residuals. Factors, transformations, conditioning, interactions, and a variety of other options are supported. The `plot.visreg()` function accepts a visreg object as calculated by `visreg()` and creates the plot.

Usage

```
## S3 method for class 'visreg'
plot(
  x,
  overlay = FALSE,
  print_cond = FALSE,
  partial = identical(x$meta$trans, I),
  band = TRUE,
  rug = 0,
  jitter = FALSE,
  strip_names = is.numeric(x$fit[, x$meta$by]),
  top = c("line", "points"),
  line = NULL,
  fill = NULL,
  points = NULL,
  gg = TRUE,
  ...
)

## S3 method for class 'visreg_list'
plot(x, ...)

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

Arguments

x	A visreg or visreg_list object; see visreg() .
overlay	By default, when by is specified, separate panels are used to display each cross-section. If overlay=TRUE, these cross-sections are overlaid on top of each other in a single plot.
print_cond	If print_cond=TRUE, the explanatory variable values conditioned on in a conditional plot are printed to the console (default: FALSE). If print_cond=TRUE and type="contrast", the conditions will still be printed, but they have no bearing on the plot unless interactions are present.
partial	If partial=TRUE (the default), partial residuals are shown on the plot.
band	If band=TRUE (the default), confidence bands are shown on the plot.
rug	Controls the appearance of a rug along the horizontal axis: • rug=0/FALSE: No rug (default) • rug=1/TRUE: Basic rug is drawn along the bottom axis • rug=2: Separate rugs drawn on the top for observations with positive residuals and on the bottom for observations with negative residuals.
jitter	If jitter=TRUE, a small amount of random noise is added to the horizontal position of partial residuals (and, if drawn, the rug). Potentially useful if many observations have exactly the same value. Only applies when xvar is continuous; for factor xvar, the horizontal spacing of points is already controlled by visreg and this argument has no effect. Default is FALSE.

strip_names	When by=TRUE, strip_names=TRUE adds the name of the by variable to the strip at the top of each panel. Default is FALSE for factors and TRUE for numeric by variables. strip_names can also be a character vector, in which case it replaces the strip names altogether with values chosen by the user.
top	By default, the fitted line is plotted on top of the partial residuals; usually this is preferable, but it does run the risk of obscuring certain residuals. To change this behavior and plot the partial residuals on top, specify top='points'.
line	List of parameters to pass to <code>ggplot2::geom_line()</code> (continuous xvar) or <code>ggplot2::geom_crossbar()</code> (factor xvar) when the fitted line is plotted.
fill	List of parameters to pass to <code>ggplot2::geom_ribbon()</code> (continuous xvar) or <code>ggplot2::geom_crossbar()</code> (factor xvar) when shaded confidence regions are plotted.
points	List of parameters to pass to <code>ggplot2::geom_point()</code> (continuous xvar) or <code>ggplot2::geom_jitter()</code> (factor xvar) when partial residuals are plotted.
gg	Deprecated, no longer has any effect.
...	Not used; present only because plot() is a generic function. An error is raised if any arguments are passed here.

Details

If x is a `visreg_list()`, `plot.visreg_list()` is dispatched instead; it simply calls `plot.visreg()` on each element in turn, passing along the same arguments to each element.

Value

A ggplot object, or if x is a `visreg_list()`, a list of ggplot objects.

References

Breheny P and Burchett W. (2017) Visualization of regression models using visreg. *R Journal*, **9**: 56-71. doi:10.32614/RJ2017046

See Also

`visreg2d` for creating visreg objects, and the `options vignette` for examples.

Examples

```
fit <- lm(Ozone ~ Solar.R + Wind + Temp, data = airquality)
visreg(fit, "Wind")

# Changing appearance
visreg(fit, "Wind", line = list(color = "red"), points = list(size = 2))
```

Description

Plot method for visualizing how two variables interact to affect the response in regression models, as a ggplot2 raster/contour plot.

Usage

```
## S3 method for class 'visreg2d'
plot(
  x,
  xlab = NULL,
  ylab = NULL,
  zlab = NULL,
  color,
  print_cond = FALSE,
  whitespace = 0.2,
  plot.type = NULL,
  ...
)
```

Arguments

x	A visreg2d() object.
xlab, ylab	Axis labels for predictors (default: variable name)
zlab	Label for the color legend (default: outcome variable name)
color	A vector of colors used to establish the color palette for the fill/legend.
print_cond	If print_cond==TRUE, the explanatory variable values conditioned on in a conditional plot are printed to the console (default: FALSE). If print_cond==TRUE and type=="contrast", the conditions will still be printed, but they have no bearing on the plot unless interactions are present.
whitespace	When xvar or yvar is a factor, whitespace determines the amount of space in between the factors. Default is 0.2, meaning that 20 percent of the axis is whitespace.
plot.type	Deprecated, no longer has any effect. See vignette .
...	Not used.

Details

For 3-dimensional surface plots, see [persp.visreg2d\(\)](#) (static) or `rgl::persp3d()` (interactive, requires the rgl package).

Value

A ggplot2 object.

References

Breheny P and Burchett W. (2017) Visualization of regression models using visreg. *R Journal*, **9**: 56-71. doi:10.32614/RJ2017046

See Also

[visreg2d\(\)](#) for creating two-dimensional visreg objects, [persp.visreg2d\(\)](#) for static 3D surface plots, and the [surface plots vignette](#) for examples and details.

Examples

```
fit <- lm(
  Ozone ~ Solar.R + Wind + Temp + I(Wind^2) + I(Temp^2) +
    I(Wind * Temp) + I(Wind * Temp^2) + I(Temp * Wind^2) + I(Temp^2 * Wind^2),
  data = airquality
)

visreg2d(fit, x = "Wind", y = "Temp")
visreg2d(fit, x = "Wind", y = "Temp", color = c("purple", "green", "red"))
```

subset.visreg	<i>Subset a visreg object</i>
---------------	-------------------------------

Description

Subset a visreg object so that only a portion of the full model is plotted.

Usage

```
## S3 method for class 'visreg'
subset(x, sub, ...)
```

Arguments

x	A visreg object.
sub	Logical expression indicating elements to keep, as in subset()
...	Not used.

Value

A [visreg\(\)](#) object.

Examples

```
# Fit a model and construct a visreg object
airquality$Heat <- cut(airquality$Temp, 3, labels = c("Cool", "Mild", "Hot"))
fit <- lm(Ozone ~ Solar.R + Wind * Heat, data = airquality)
v <- visreg(fit, "Wind", by = "Heat", plot = FALSE)

# Plot only certain levels
subset(v, Heat %in% c("Cool", "Hot")) |> plot()

# Plot only up to wind 15 mph
subset(v, Wind < 15) |> plot()
```

visreg

Visualization of regression functions

Description

A function for visualizing regression models quickly and easily. Default plots contain a confidence band, prediction line, and partial residuals. Factors, transformations, conditioning, interactions, and a variety of other options are supported. The `visreg` function performs the calculations and, if `plot=TRUE` (the default), these calculations are passed to `plot.visreg` for plotting.

Usage

```
visreg(
  fit,
  xvar,
  by,
  breaks = 3,
  type = c("conditional", "contrast"),
  data = NULL,
  trans = I,
  scale = c("linear", "response"),
  alpha = 0.05,
  nn = 101,
  cond = list(),
  collapse = FALSE,
  plot = TRUE,
  predict = list(),
  ...
)
```

Arguments

<code>fit</code>	The fitted model object you wish to visualize. Any object with 'predict' and 'model.frame' methods are supported, including <code>lm</code> , <code>glm</code> , <code>gam</code> , <code>rlm</code> , <code>coxph</code> , and many more.
------------------	--

xvar	Character string specifying the variable to be put on the x-axis of your plot. Both continuous variables and factors are supported.
by	(Optional) A variable allowing you to divide your plot into cross-sections based on levels of the by variable; particularly useful for visualizing models with interactions. Supplied as a character string. Cross-sections are displayed as facets of a ggplot2 plot. Both continuous variables and factors are supported.
breaks	If a continuous variable is used for the by option, the breaks argument controls the values at which the cross-sections are taken. By default, cross-sections are taken at the 10th, 50th, and 90th quantiles. If breaks is a single number, it specifies the number of breaks. If breaks is a vector of numbers, it specifies the values at which the cross-sections are to be taken. Each partial residuals appears exactly once in the plot, in the panel it is closest to.
type	The type of plot to be produced. The following options are supported: • If "conditional" is selected, the plot returned shows the value of the variable on the x-axis and the change in response on the y-axis, holding all other variables constant (by default, median for numeric variables and most common category for factors). • If "contrast" is selected, the plot returned shows the effect on the expected value of the response by moving the x variable away from a reference point on the x-axis (for numeric variables, this is taken to be the mean). For more details, see references.
data	The data frame used to fit the model. Typically, visreg() can figure out where the data is, so it is not necessary to provide this. In some cases, however, the data set cannot be located and must be supplied explicitly.
trans	(Optional) A function specifying a transformation for the vertical axis.
scale	By default, the model is plotted on the scale of the linear predictor. If scale='response' for a glm, the inverse link function will be applied so that the model is plotted on the scale of the original response.
alpha	Alpha level (1-coverage) for the confidence band displayed in the plot (default: 0.05).
nn	Controls the smoothness of the line and confidence band. Increasing this number will add to the computational burden, but produce a smoother plot (default: 101).
cond	Named list specifying conditional values of other explanatory variables. By default, conditional plots in visreg are constructed by filling in other explanatory variables with the median (for numeric variables) or most common category (for factors), but this can be overridden by specifying their values using cond (see examples).
collapse	If the predict method for fit returns a matrix, should this be returns as multiple visreg objects bound together as a list (collapse=FALSE) or collapsed down to a single visreg object (collapse=TRUE).
plot	Send the calculations to <code>plot.visreg()</code> ? Default is TRUE.
predict	Named list of additional arguments to pass to predict() (e.g., list(re.form = NA) for a mixed model). Most models never need this; it exists as an escape hatch since predict() methods vary by model class and aren't otherwise accessible from visreg().

... Graphical parameters (e.g., partial) to pass to `plot.visreg()` (see examples below).

Details

See `plot.visreg()` for plotting options, such as changing the appearance of points, lines, confidence bands, etc.

Value

A `visreg` or `visreg_list` object (which is simply a list of `visreg` objects). A `visreg` object has three components:

- fit** A data frame with `nn` rows containing the fit of the model as `xvar` varies, along with lower and upper confidence bounds (named `visreg_fit`, `visreg_lwr`, and `visreg_upr`, respectively). The fitted matrix of coefficients.
- res** A data frame with `n` rows, where `n` is the number of observations in the original data set used to model. This frame contains information about the residuals, named `visregReg` and `visreg_pos`; the latter records whether the residual was positive or negative.
- meta** Contains meta-information needed to construct plots, such as the name of the `x` and `y` variables, whether there were any by variables, etc.

Note that if `plot = TRUE` (the default), then the `visreg` object is passed to `plot.visreg()` and a `ggplot` object is returned instead.

References

- <https://pbreheny.github.io/visreg/>
- Breheny P and Burchett W. (2017) Visualization of regression models using visreg. *R Journal*, 9: 56–71. doi:10.32614/RJ2017046

See Also

`plot.visreg()` for plotting options, `visreg2d()` for creating two-dimensional `visreg` objects, and the [package website](#) for detailed explanations and examples.

Examples

```
library(ggplot2)

# --- Linear models -----

## Basic
fit <- lm(Ozone ~ Solar.R + Wind + Temp, data = airquality)
visreg(fit, "Wind", type = "contrast")
visreg(fit, "Wind", type = "conditional")

## Factors
airquality$Heat <- cut(airquality$Temp, 3, labels = c("Cool", "Mild", "Hot"))
fit.heat <- lm(Ozone ~ Solar.R + Wind + Heat, data = airquality)
```

```

visreg(fit.heat, "Heat", type = "contrast")
visreg(fit.heat, "Heat", type = "conditional")

## Transformations
fit1 <- lm(Ozone ~ Solar.R + Wind + Temp + I(Wind^2), data = airquality)
fit2 <- lm(log(Ozone) ~ Solar.R + Wind + Temp, data = airquality)
fit3 <- lm(log(Ozone) ~ Solar.R + Wind + Temp + I(Wind^2), data = airquality)
visreg(fit1, "Wind")
visreg(fit2, "Wind", trans = exp) + ylab("Ozone")
visreg(fit3, "Wind", trans = exp) + ylab("Ozone")

## Conditioning
visreg(fit, "Wind", cond = list(Temp = 50))
visreg(fit, "Wind", print_cond = TRUE)
visreg(fit, "Wind", cond = list(Temp = 100))

## Interactions
fit_int <- lm(Ozone ~ Solar.R + Wind * Heat, data = airquality)
visreg(fit_int, "Wind", by = "Heat")
visreg(fit_int, "Heat", by = "Wind")
visreg(fit_int, "Wind", by = "Heat", type = "contrast")
visreg(fit_int, "Heat", by = "Wind", breaks = 6)
visreg(fit_int, "Heat", by = "Wind", breaks = c(0, 10, 20))

## Overlay
visreg(fit_int, "Wind", by = "Heat", overlay = TRUE)

# --- Nonlinear models -----

## Logistic regression
data("birthwt", package = "MASS")
birthwt$race <- factor(birthwt$race, labels = c("White", "Black", "Other"))
birthwt$smoke <- factor(birthwt$smoke, labels = c("Nonsmoker", "Smoker"))
fit <- glm(low ~ age + race + smoke + lwt, data = birthwt, family = "binomial")
visreg(fit, "lwt") +
  xlab("Mother's Weight") +
  ylab("Log odds (low birthweight)")
visreg(fit, "lwt", scale = "response", partial = FALSE) +
  xlab("Mother's Weight") +
  ylab("P(low birthweight)")

## Proportional hazards
data(cancer, package="survival")
ovarian$rx <- factor(ovarian$rx)
fit <- survival::coxph(
  survival::Surv(futime, fustat) ~ age + rx,
  data = ovarian
)
visreg(fit, "age") + ylab("log(Hazard ratio)")

## Robust regression
fit <- MASS::rlm(Ozone ~ Solar.R + Wind * Heat, data = airquality)

```

```
visreg(fit, "Wind", cond = list(Heat = "Mild"))

## And more...; anything with a 'predict' method should work

## Return raw components of plot
v <- visreg(fit, "Wind", cond = list(Heat = "Mild"))
```

visreg2d

*Visualization of regression functions for two variables***Description**

A function used to visualize how two variables interact to affect the response in regression models.

Usage

```
visreg2d(
  fit,
  xvar,
  yvar,
  type = c("conditional", "contrast"),
  data = NULL,
  trans = I,
  scale = c("linear", "response"),
  nn = 99,
  cond = list(),
  plot = TRUE,
  ...
)
```

Arguments

- | | |
|------|--|
| fit | The fitted model object you wish to visualize. Any object with 'predict' and 'model.frame' methods are supported, including lm, glm, gam, rlm, coxph, and many more. |
| xvar | Character string specifying the variable to be put on the x-axis of your plot. Both continuous variables and factors are supported. |
| yvar | Character string specifying the variable to be put on the y-axis of your plot. Both continuous variables and factors are supported. |
| type | <p>The type of plot to be produced. The following options are supported:</p> <ul style="list-style-type: none"> • If "conditional" is selected, the plot returned shows the value of the variable on the x-axis and the change in response on the y-axis, holding all other variables constant (by default, median for numeric variables and most common category for factors). |

	• If "contrast" is selected, the plot returned shows the effect on the expected value of the response by moving the x variable away from a reference point on the x-axis (for numeric variables, this is taken to be the mean). For more details, see references.
data	The data frame used to fit the model. Typically, visreg() can figure out where the data is, so it is not necessary to provide this. In some cases, however, the data set cannot be located and must be supplied explicitly.
trans	(Optional) A function specifying a transformation for the vertical axis.
scale	By default, the model is plotted on the scale of the linear predictor. If scale= 'response' for a glm, the inverse link function will be applied so that the model is plotted on the scale of the original response.
nn	Resolution of the three dimensional plot. Higher values will results in a smoother looking plot.
cond	Named list specifying conditional values of other explanatory variables. By default, conditional plots in visreg are constructed by filling in other explanatory variables with the median (for numeric variables) or most common category (for factors), but this can be overridden by specifying their values using cond (see examples).
plot	Send the calculations to <code>plot.visreg2d()</code> , producing a ggplot2 raster/contour plot? Default is TRUE. For a static 3D perspective plot or an interactive rgl plot, use plot = FALSE and pass the result to <code>persp.visreg2d()</code> or <code>rgl::persp3d()</code> instead; see examples.
...	Graphical parameters (e.g., ylab) can be passed to the function to customize the plots.

Value

A visreg2d object consisting of:

x Values of xvar to be plotted

y Values of yvar to be plotted

z Values of outcome to be plotted

meta Meta-information needed to construct plots, such as the name of the x and y variables.

References

- <https://pbreheny.github.io/visreg/>
- Breheny P and Burchett W. (2017) Visualization of regression models using visreg. *R Journal*, 9: 56-71. doi:10.32614/RJ2017046

See Also

`plot.visreg2d()` for the 2-dimensional raster/contour plot, `persp.visreg2d()` for a static 3-dimensional perspective plot, and the [vignette](#) for examples.

Examples

```
fit <- lm(
  Ozone ~ Solar.R + Wind + Temp + I(Wind^2) + I(Temp^2) +
    I(Wind * Temp) + I(Wind * Temp^2) + I(Temp * Wind^2) + I(Temp^2 * Wind^2),
  data = airquality
)

visreg2d(fit, x = "Wind", y = "Temp")
visreg2d(fit, x = "Wind", y = "Temp", plot = FALSE) |> persp()

## Requires the rgl package
## Not run:
visreg2d(fit, x = "Wind", y = "Temp", plot = FALSE) |> rgl::persp3d()

## End(Not run)
```

visreg_list

Join multiple visreg objects together in a list

Description

This function takes multiple visreg objects, from separate calls to [visreg\(\)](#), and joins them together in a single object. The single object will be of type visreg_list unless collapse=TRUE is specified, in which case the list will be collapsed back down into a single visreg object.

Usage

```
visreg_list(..., labels, collapse = FALSE)
```

Arguments

...	visreg objects, as produced by calls to visreg() .
labels	A character vector with length corresponding to the number of visreg objects passed to the function that provides labels for the different objects in subsequent plots. Only has an effect if collapse=TRUE.
collapse	If TRUE, the resulting object will be collapsed down into a single visreg object. If FALSE, the resulting object will be a visreg_list.

Value

A visreg or visreg_list object, depending on the value of collapse.

See Also

[visreg\(\)](#), [plot.visreg\(\)](#)

Examples

```
fit <- lm(Ozone ~ Solar.R + Wind + Temp, data = airquality)
v1 <- visreg(fit, "Wind", plot = FALSE, alpha = 0.2)
v2 <- visreg(fit, "Wind", plot = FALSE, alpha = 0.01)
vv1 <- visreg_list(v1, v2, collapse = FALSE)
vv2 <- visreg_list(v1, v2,
  collapse = TRUE,
  labels = c("Confidence: 0.80", "Confidence: 0.99")
)
plot(vv1)
plot(vv2)
```

Index

`collapse`, [2](#)

`ggplot2::geom_crossbar()`, [6](#)
`ggplot2::geom_jitter()`, [6](#)
`ggplot2::geom_line()`, [6](#)
`ggplot2::geom_point()`, [6](#)
`ggplot2::geom_ribbon()`, [6](#)
`graphics::persp()`, [3](#), [4](#)

`persp.visreg2d`, [3](#)
`persp.visreg2d()`, [7](#), [8](#), [14](#)
`persp.visreg_list` (`persp.visreg2d`), [3](#)
`persp3d.visreg_list` (`persp.visreg2d`), [3](#)
`plot.visreg`, [4](#)
`plot.visreg()`, [10](#), [11](#), [15](#)
`plot.visreg2d`, [7](#)
`plot.visreg2d()`, [4](#), [14](#)
`plot.visreg_list` (`plot.visreg`), [4](#)
`print.visreg_plot_list` (`plot.visreg`), [4](#)

`subset()`, [8](#)
`subset.visreg`, [8](#)

`visreg`, [9](#)
`visreg()`, [2](#), [4](#), [5](#), [8](#), [15](#)
`visreg2d`, [6](#), [13](#)
`visreg2d()`, [3](#), [4](#), [7](#), [8](#), [11](#)
`visreg_list`, [15](#)
`visreg_list()`, [2](#), [6](#)