Package 'visreg'

August 20, 2025

Title Visualization of Regression Models

Version 2.8.0

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 lattice

Suggests ggplot2, glmmTMB, knitr, lme4, MASS, Matrix, rgl, rmarkdown,
survival, tinytest

Enhances nlme

VignetteBuilder knitr

Encoding UTF-8

RoxygenNote 7.3.2

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 2025-08-20 15:00:02 UTC

Contents

---

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 plot.visreg() function accepts a visreg or visregList object as calculated by [visreg()](#) and creates the plot.

Usage

```
## S3 method for class 'visreg'
plot(
  x,
  overlay = FALSE,
  print.cond = FALSE,
  whitespace = 0.2,
  partial = identical(x$meta$trans, I),
  band = TRUE,
  rug = ifelse(partial, 0, 2),
  strip.names = is.numeric(x$fit[, x$meta$by]),
  legend = TRUE,
  top = c("line", "points"),
  gg = FALSE,
  line.par = NULL,
  fill.par = NULL,
  points.par = NULL,
  ...
)
```

Arguments

x	A visreg or visregList 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.
whitespace	When xvar is a factor, whitespace determines the amount of space in between factors on the x-axis. Default is 0.2, meaning that 20 percent of the horizontal axis is whitespace.
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	By default, partial residuals are plotted. Alternatively, a [rug()](rug()) may be plotted along the horizontal axis instead. Setting rug=TRUE turns off partial residuals by default; if one wants both to be plotted, both rug=TRUE and partial=TRUE need to be specified. Two types of rug plots are available. If rug=1 or rug=TRUE, then a basic rug is drawn on the bottom. If rug=2, then separate rugs are drawn on the top for observations with positive residuals and on the bottom for observations with negative residuals. Such plots are particularly useful in logistic regression (see examples).
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.
legend	For overlay plots, (overlay=TRUE), should visreg create a legend? If legend=TRUE (the default), a legend is placed in the top margin.
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'.
gg	By default (gg=FALSE), visreg will use the **lattice** package to render the plot if multiple panels are required. If gg=TRUE, it will use the **ggplot2** package instead, provided that it is installed.
line.par	List of parameters (see [par()](par())) to pass to lines(...) or ggplot2::geom_line() when lines are plotted.
fill.par	List of parameters (see [par()](par())) to pass to polygon(...) or ggplot2::geom_polygon() when shaded confidence regions are plotted.
points.par	List of parameters ([par()](par())) to pass to points(...) or ggplot2::geom_point() when partial residuals are plotted.
...	Graphical parameters can be passed to the function to customize the plots. If by=TRUE, lattice parameters can be passed, such as layout (see examples below).

Author(s)

Patrick Breheny and Woodrow Burchett

References

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

See Also

https://pbreheny.github.io/visreg/articles/options.html, visreg(), visreg2d()

Examples

```
fit <- lm(Ozone ~ Solar.R + Wind + Temp,data=airquality)
visreg(fit, "Wind", line=list(col="red"), points=list(cex=1, pch=1))

## Changing appearance
visreg(fit, "Wind", line=list(col="red"), points=list(cex=1, pch=1))

## See ?visreg and https://pbreheny.github.io/visreg for more examples
```

---

plot.visreg2d   *Visualization of regression functions for two variables*

Description

Plot method for visualizing how two variables interact to affect the response in regression models.

Usage

```
## S3 method for class 'visreg2d'
plot(
  x,
  plot.type = c("image", "persp", "rgl", "gg"),
  xlab,
  ylab,
  zlab,
  color,
  print.cond = FALSE,
  whitespace = 0.2,
  ...
)
```

Arguments

x	A visreg2d() object.
plot.type	The style of plot to be produced. The following options are supported:
	• image: a filled contour
	• gg: a filled contour plot using ggplot2
	• persp: a 3 dimensional perspective plot
	• rgl: a version of the perspective plot that can be rotated (requires the rgl package to be installed)
xlab	Axis label for x variable
ylab	Axis label for y variable

zlab	Axis label for outcome
color	For `plot.type='persp'` or `plot.type='rgl'`, the color of the surface. For `plot.type='image'` or `plot.type='gg'`, a vector of colors used to establish a color palette.
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.
...	Graphical parameters can be passed to the function to customize the plots.

Author(s)

Patrick Breheny and Woodrow Burchett

References

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

See Also

https://pbreheny.github.io/visreg/surface.html, visreg()

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", plot.type="image")
visreg2d(fit, x="Wind", y="Temp", plot.type="image",
         color=c("purple", "green", "red"))
visreg2d(fit, x="Wind", y="Temp", plot.type="persp")

## Requires the rgl package

visreg2d(fit,x="Wind",y="Temp",plot.type="rgl")


## Requires the ggplot2 package

visreg2d(fit, x="Wind", y="Temp", plot.type="gg")
```

---

subset.visreg   *Subset a visreg object*

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.

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
vv <- subset(v, Heat %in% c("Cool", "Hot"))
plot(vv)

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

---

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"),
  xtrans,
  alpha = 0.05,
  nn = 101,
  cond = list(),
  jitter = FALSE,
  collapse = FALSE,
  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.
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. Uses the lattice package. 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.
xtrans	(Optional) A function specifying a transformation for the horizontal axis. Note that, for model terms such as log(x), visreg automatically plots on the original axis (see examples).
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).
jitter	Adds a small amount of noise to xvar. Potentially useful if many observations have exactly the same value. Default is FALSE.
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 plot.visreg? Default is TRUE.
...	Graphical parameters (e.g., ylab) can be passed to the function to customize the plots. If by=TRUE, lattice parameters can be passed, such as layout (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 visregList 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 visregFit, visregLwr, and visregUpr, 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 visregPos; 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.

Author(s)

Patrick Breheny and Woodrow Burchett

References

- <https://pbreheny.github.io/visreg/>
- Breheny, P. and Burchett, W. (2017), Visualizing regression models using visreg. <https://journal.r-project.org/archive/2017/RJ-2017-046/index.html>

See Also

<https://pbreheny.github.io/visreg/> [plot.visreg()] [visreg2d)]

[visreg2d)]: R:visreg2d)

Examples

```
# --- 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.in1 <- lm(Ozone~ Solar.R + Wind*Heat, data=airquality)
visreg(fit.in1, "Wind", by="Heat")
visreg(fit.in1, "Heat", by="Wind")
visreg(fit.in1, "Wind", by="Heat", type="contrast")
```

```
visreg(fit.in1, "Heat", by="Wind", breaks=6)
visreg(fit.in1, "Heat", by="Wind", breaks=c(0,10,20))

## Overlay
visreg(fit.in1, "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)")
visreg(fit, "lwt", scale="response", partial=FALSE,
       xlab="Mother's Weight", ylab="P(low birthweight)", rug=2)

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

## Robust regression
require(MASS)
fit <- 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	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.
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 [plot.visreg2d()], producing a plot? Default is TRUE.

...	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.

Author(s)

Patrick Breheny and Woodrow Burchett

References

- <https://pbreheny.github.io/visreg/>
- Breheny, P. and Burchett, W. (2017), Visualizing regression models using visreg. https://journal.r-project.org/archive/2017/RJ-2017-046/index.html

See Also

<https://pbreheny.github.io/visreg/articles/surface.html> [visreg()]

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", plot.type="image")
visreg2d(fit, x="Wind", y="Temp", plot.type="persp")

## Requires the rgl package
## Not run:
visreg2d(fit, x="Wind", y="Temp", plot.type="rgl")

## End(Not run)

## Requires the ggplot2 package
## Not run:
visreg2d(fit, x="Wind", y="Temp", plot.type="gg")

## End(Not run)
```

---

visregList   *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 `visregList` unless `collapse=TRUE` is specified, in which case the list will be collapsed back down into a single `visreg` object.

Usage

```
visregList(..., 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 `visregList`.

Value

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

Author(s)

Patrick Breheny

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 <- visregList(v1, v2, collapse=FALSE)
vv2 <- visregList(v1, v2, collapse=TRUE,
                  labels=c("Confidence: 0.80", "Confidence: 0.99"))
op <- par(mfrow=c(1,2))
plot(vv1)
par(op)
plot(vv2)
```

Index