Package ‘oddsratio’

October 14, 2022

Title Odds Ratio Calculation for GAM(M)s & GLM(M)s

Version 2.0.1

Description  Simplified odds ratio calculation of GAM(M)s &
             GLM(M)s. Provides structured output (data frame) of all predictors and
             their corresponding odds ratios and confident intervals for further
             analyses. It helps to avoid false references of predictors and
             increments by specifying these parameters in a list instead of using
             'exp(coef(model))' (standard approach of odds ratio calculation for
             GLMs) which just returns a plain numeric output. For GAM(M)s, odds
             ratio calculation is highly simplified with this package since it
             takes care of the multiple 'predict()' calls of the chosen predictor
             while holding other predictors constant. Also, this package allows
             odds ratio calculation of percentage steps across the whole predictor
             distribution range for GAM(M)s. In both cases, confident intervals
             are returned additionally. Calculated odds ratio of GAM(M)s can be
             inserted into the smooth function plot.

License MIT + file LICENSE

URL https://github.com/pat-s/oddsratio

BugReports https://github.com/pat-s/oddsratio/issues

Depends R (>= 3.0.0)

Imports ggplot2 (>= 3.0.0), mgcv, stats,

Suggests knitr, MASS, rmarkdown, testthat

VignetteBuilder knitr

Encoding UTF-8

LazyData true

RoxygenNote 7.1.0

NeedsCompilation no

Author Patrick Schratz [aut, cre] (<https://orcid.org/0000-0003-0748-6624>)

Maintainer Patrick Schratz <patrick.schratz@gmail.com>

Repository CRAN

Date/Publication 2020-05-24 22:00:02 UTC
Description
This function inserts calculated odds ratios of GAM(M)s into a plot of a GAM(M) smoothing function.

Usage

```r
insert_or(
  plot_object = NULL,
  or_object = NULL,
  line_col = "red",
  line_size = 1.2,
  line_type = "solid",
  line_alpha = 1,
  text_alpha = 1,
  text_size = 4,
  text_col = "black",
  rect_alpha = 0.5,
  rect_col = NULL,
  rect = FALSE,
  arrow = TRUE,
  values = TRUE,
  values_yloc = 0,
  values_xloc = NULL,
  or_yloc = 0,
  arrow_length = NULL,
  arrow_yloc = NULL,
  arrow_col = NULL,
  arrow_xloc_r = NULL,
  arrow_xloc_l = NULL
)
```

Arguments

- `plot_object`: A ggplot object from `plot_gam()`.
- `or_object`: A data.frame as returned from `or_gam()`.
The idea behind this function is to add calculated odds ratios of fitted GAM models (or_gam()) into a plot showing the smooth function (plot_gam) of the chosen predictor for which the odds ratio was calculated for. Multiple insertions can be made by iterative calling the function (see examples).

Right now the function only accepts inputs from or_gam() objects with slice = FALSE. If you want to insert multiple odds ratio values, call the function multiple times.

Value
ggplot2

See Also
plot_gam(), or_gam()

Examples

library(oddsratio)
library(mgcv)
fit_gam <- gam(y ~ s(x0) + s(I(x1^2)) + s(x2) +
offset(x3) + x4, data = data_gam) # fit model

# create input objects (plot + odds ratios)
plot_object <- plot_gam(fit_gam, pred = "x2", title = "Predictor 'x2'")
or_object1 <- or_gam(  
  data = data_gam, model = fit_gam,
  pred = "x2", values = c(0.099, 0.198)
)

# insert first odds ratios to plot
plot_object <- insert_or(plot_object, or_object1,
or_gam

Calculate Odds Ratios of Generalized Additive (Mixed) Models

Description

This function calculates odds ratio(s) for specific increment steps of GAM(M) models. Odds ratios can also be calculated for continuous (percentage) increment steps across the whole predictor distribution using slice = TRUE.

Usage

or_gam(
  data = NULL,
  model = NULL,
  pred = NULL,
  values = NULL,
  percentage = NULL,
  slice = FALSE,
  ci = NULL
)

Arguments

data The data used for model fitting.
model A fitted GAM(M).
pred Predictor name for which to calculate the odds ratio.
or_gam

values  Numeric vector of length two. Predictor values to estimate odds ratio from. Function is written to use the first provided value as the "lower" one, i.e. calculating the odds ratio 'from value1 to value2'. Only used if slice = FALSE.

percentage  Percentage number to split the predictor distribution into. A value of 10 would split the predictor distribution by 10%. Only needed if slice = TRUE.

slice  Whether to calculate odds ratios for fixed increment steps over the whole predictor distribution. See percentage for setting the increment values.

Value  A data.frame with (up to) eight columns.perc1 and perc2 are only returned if slice = TRUE:

predictor  Predictor name
value1  First value of odds ratio calculation
value2  Second value of odds ratio calculation
perc1  Percentage value of value1
perc2  Percentage value of value2
oddsratio  Calculated odds ratio(s)
.ci_low  Lower (2.5%) confident interval of odds ratio
.ci_high  Higher (97.5%) confident interval of odds ratio

See Also  or_glm() plot_gam() insert_or()

Examples

library(oddsratio)
library(mgcv)
fit_gam <- gam(y ~ s(x0) + s(I(x1^2)) + s(x2) + offset(x3) + x4, data = data_gam)  # fit model

# Calculate OR for specific increment step of continuous variable
or_gam(
  data = data_gam, model = fit_gam, pred = "x2",
  values = c(0.099, 0.198)
)

## Calculate OR for change of indicator variable
or_gam(
  data = data_gam, model = fit_gam, pred = "x4",
  values = c("B", "D")
)

## Calculate ORs for percentage increments of predictor distribution
## or_glm

### (here: 20%)

```r
or_gam(
  data = data_gam, model = fit_gam, pred = "x2",
  percentage = 20, slice = TRUE
)
```

### or_glm

**Calculate Odds Ratios of Generalized Linear (Mixed) Models**

### Description

This function calculates odds ratio(s) for specific increment steps of GLMs.

### Usage

```r
or_glm(data, model, incr, ci = 0.95)
```

### Arguments

- **data**: The data used for model fitting.
- **model**: A fitted GLM(M).
- **incr**: Increment values of each predictor given in a named list.
- **ci**: Which confidence interval to calculate. Must be between 0 and 1. Default to 0.95

### Details

`ci_low` and `ci_high` are only calculated for GLM models because `MASS::glmmPQL()` does not return confident intervals due to its penalizing behavior.

Currently supported functions: `stats::glm, MASS::glmmPQL`

### Value

A data frame with five columns:

- **predictor**: Predictor name(s)
- **oddsratio**: Calculated odds ratio(s)
- **ci_low**: Lower confident interval of odds ratio
- **ci_high**: Higher confident interval of odds ratio
- **increment**: Increment of the predictor(s)

### See Also

`or_gam()`
### Examples

```r
## Example with glm()
library(oddsratio)
# load data (source: http://www.ats.ucla.edu/stat/r/dae/logit.htm) and 
# fit model
fit_glm <- glm(admit ~ gre + gpa + rank,
               data = data_glm,
               family = "binomial"
) # fit model

# Calculate OR for specific increment step of continuous variable
or_glm(data = data_glm, model = fit_glm, incr = list(gre = 380, gpa = 5))

# Calculate OR and change the confidence interval level
or_glm(
  data = data_glm, model = fit_glm,
  incr = list(gre = 380, gpa = 5), ci = .70
)

## Example with MASS:glmmPQL()
# load data
library(MASS)
data(bacteria)
fit_glmmPQL <- glmmPQL(y ~ trt + week,
                    random = ~ 1 | ID,
                    family = binomial, data = bacteria,
                    verbose = FALSE
)

# Apply function
or_glm(data = bacteria, model = fit_glmmPQL, incr = list(week = 5))
```

---

**plot_gam**  

*Plot GAM(M) Smoothing Function*

**Description**

Plots the smoothing function of GAM(M) predictors via ggplot2

**Usage**

```r
plot_gam(
  model = NULL,
  pred = NULL,
  col_line = "blue",
  ci_line_col = "black",
  ci_line_type = "dashed",
  ci_fill = "grey",
  ci_alpha = 0.4, 
)```

ci_line_size = 0.8,
sm_fun_size = 1.1,
title = NULL,
xlab = NULL,
ylab = NULL,
limits_y = NULL,
breaks_y = NULL
)

Arguments

- **model**: A fitted model of class `gam`.
- **pred**: Predictor name.
- **col_line**: Smoothing function line color.
- **ci_line_col**: Confident interval line color.
- **ci_line_type**: Linetype of confidence interval.
- **ci_fill**: Fill color of area between smoothing function and its confidence interval lines.
- **ci_alpha**: Opacity value of confidence interval.
- **ci_line_size, sm_fun_size**: Line sizes.
- **title**: Plot title.
- **xlab**: x-axis title.
- **ylab**: y-axis title.
- **limits_y**: y-axis limits.
- **breaks_y**: y-axis breaks. Values are handed over to a `seq` call, e.g. `seq(-6, 6, 2).

See Also

- `or_gam()` `insert_or()`

Examples

```r
library(oddsratio)
library(mgcv)
fit_gam <- mgcv::gam(y ~ s(x0) + s(I(x1^2)) + s(x2) + offset(x3) + x4,
data = data_gam
)
plot_gam(fit_gam, pred = "x2", title = "Predictor 'x2'")
```
Index

data.frame, 2, 5
gam::gam, 5
ggplot2, 3

insert_or, 2
insert_or(), 5, 8

MASS::glmmPQL, 6
MASS::glmmPQL(), 6
mgcv::gam, 5
mgcv::gamm, 5

or_gam, 4, 5
or_gam(), 2, 3, 6, 8
or_glm, 6
or_glm(), 5

plot_gam, 3, 7
plot_gam(), 2, 3, 5

stats::glm, 6