

# Package ‘oddsratio’

July 25, 2018

**Version** 1.0.3

**Date** 2018-07-15

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

**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.

**URL** <https://github.com/pat-s/oddsratio>

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

**Depends** R (>= 3.0.0)

**Imports** mgcv, MASS, stats, ggplot2 (>= 3.0.0), cowplot

**License** MIT + file LICENSE

**RoxygenNote** 6.0.1

**Suggests** knitr, rmarkdown, grid, gtable, scales, testthat, pacman, gam

**LazyData** true

**ByteCompile** true

**VignetteBuilder** knitr

**NeedsCompilation** no

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

**Maintainer** Patrick Schratz <[patrick.schratz@gmail.com](mailto:patrick.schratz@gmail.com)>

**Repository** CRAN

**Date/Publication** 2018-07-25 12:20:02 UTC

## R topics documented:

insert_or . . . . .	2
or_gam . . . . .	4
or_glm . . . . .	6
plot_gam . . . . .	7

<b>Index</b>	<b>9</b>
--------------	----------

---

insert_or	<i>Insert odds ratios of GAM(M)s into smoothing function</i>
-----------	--

---

### Description

This function inserts calculated odds ratios of GAM(M)s into a plot of a GAM(M) smoothing function.

### Usage

```
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 <a href="#">plot_gam</a> .
or_object	A returned data.frame from <a href="#">or_gam</a> .
line_col, line_alpha, line_type, line_size	Aesthetics of vertical lines.
text_col, text_alpha, text_size	Aesthetics of inserted values.
rect_col, rect_alpha	Aesthetics of shaded rectangle.
rect	Logical. Whether to print a shaded rectangle between the vertical lines.
arrow	Logical. Whether to print arrows above the inserted values. Default to TRUE.
values	Logical. Whether to print predictor value information nearby the inserted vertical lines. Default to TRUE.
values_xloc	Numeric. X-axis location/shift of values relative to their vertical line. Default to 2% of x-axis range.
or_yloc, values_yloc	Numeric. Specifies y-location of inserted odds ratio / values. Relative to plotted y-axis range. A positive/negative value will place the the text higher/lower.

arrow\_xloc\_r, arrow\_xloc\_l, arrow\_yloc, arrow\_length, arrow\_col  
 Numeric. Axis placement options of inserted arrows. Relative to respective axis ranges.

### Details

The idea behind this function is to add calculated odds ratio 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 iteratively calling the function (see examples).

Right now the function does only accept results of [or\\_gam](#) with `slice = FALSE`. If you want to insert multiple odds ratio you have to do it iteratively.

### Value

Returns a ggplot plotting object

### Author(s)

Patrick Schratz [patrick.schratz@gmail.com](mailto:patrick.schratz@gmail.com)

### See Also

[plot\\_gam](#)

[or\\_gam](#)

### Examples

```
# load data (Source: ?mgcv::gam) and fit model
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)
library(oddsratio)
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_yloc = 3,
                        values_xloc = 0.04, line_size = 0.5,
                        line_type = "dotdash", text_size = 6,
                        values_yloc = 0.5, arrow_col = "red")

# calculate second odds ratio
or_object2 <- or_gam(data = data_gam, model = fit_gam, pred = "x2",
                    values = c(0.4, 0.6))

# add or_object2 into plot
insert_or(plot_object, or_object2, or_yloc = 2.1, values_yloc = 2,
          line_col = "green4", text_col = "black",
```

```
rect_col = "green4", rect_alpha = 0.2,
line_alpha = 1, line_type = "dashed",
arrow_xloc_r = 0.01, arrow_xloc_l = -0.01,
arrow_length = 0.01, rect = TRUE)
```

---

or\_gam

---

*Calculate odds ratios of Generalized Additive (Mixed) Models*


---

### Description

This function calculates odds ratio(s) for specific increment steps of a GAM(M)s.

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

<code>data</code>	The data used for model fitting.
<code>model</code>	A fitted GAM(M).
<code>pred</code>	Character. Predictor name for which to calculate the odds ratio.
<code>values</code>	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 <code>slice = FALSE</code> .
<code>percentage</code>	Numeric. Percentage number to split the predictor distribution into. A value of 10 would split the predictor distribution by 10% intervals. Only needed if <code>slice = TRUE</code> .
<code>slice</code>	Logical. Default = <code>FALSE</code> . Whether to calculate odds ratios for fixed increment steps over the whole predictor distribution. See <code>percentage</code> for setting the increment values.
<code>CI</code>	Numeric. Currently fixed to 95% confidence interval level (2.5% - 97.5%). It should not be changed in a function call!

### Details

Currently supported functions: [mgcv::gam](#), [mgcv::gamm](#), [gam::gam](#).

For [mgcv::gamm](#), the model input of `or_gam` needs to be the gam output (e.g. `fit_gam$gam`).

**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

**Author(s)**

Patrick Schratz [patrick.schratz@gmail.com](mailto:patrick.schratz@gmail.com)

**See Also**

[or\\_glm](#)

[plot\\_gam](#)

[insert\\_or](#)

**Examples**

```
# load data (Source: ?mgcv::gam) and fit model
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
## (here: 20%)
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**

```
or_glm(data, model, incr, CI = 0.95)
```

**Arguments**

<code>data</code>	The data used for model fitting.
<code>model</code>	A fitted GLM(M).
<code>incr</code>	List. Increment values of each predictor.
<code>CI</code>	numeric. Which confident 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 [glmmPQL](#) does not return confident intervals due to its penalizing behavior.

Currently supported functions: [glm](#), [glmmPQL](#)

**Value**

A data frame with five columns:

<code>predictor</code>	Predictor name(s)
<code>oddsratio</code>	Calculated odds ratio(s)
<code>CI_low</code>	Lower confident interval of odds ratio
<code>CI_high</code>	Higher confident interval of odds ratio
<code>increment</code>	Increment of the predictor(s)

**Author(s)**

Patrick Schratz [patrick.schratz@gmail.com](mailto:patrick.schratz@gmail.com)

**See Also**

[or\\_gam](#)

**Examples**

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

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

---

plot\_gam

*Plot smoothing functions of GAM(M) models*


---

**Description**

This function plots the smoothing function of selected GAM(M) models using the ggplot2 plotting system.

**Usage**

```
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	The predictor of the fitted model to plot the smooth function of.
col_line	Character. Sets color for smoothing function. Default to "blue".
ci_line_col	Character. Sets color for confident interval line of smoothing function. Default to "black".

ci_line_type	Character. Sets linetype of confident interval line of smoothing function. Default to "dashed".
ci_fill	Character. Fill color of area between smoothing function and its confident interval lines.
ci_alpha	Numeric (range: 0-1). Opacity value of confidence interval shading.
ci_line_size, sm_fun_size	Line sizes.
title	Character. Plot title.
xlab	Character. X-axis title.
ylab	Character. Y-axis title.
limits_y	Numeric of length two. Sets y-axis limits.
breaks_y	Numeric of length three. Sets y-axis breaks. See <a href="#">seq</a> . Values need to be given in a seq call, e.g. seq(-6, 6, 2).

**Author(s)**

Patrick Schratz [patrick.schratz@gmail.com](mailto:patrick.schratz@gmail.com)

**See Also**

[plot\\_gam](#)

[or\\_gam](#)

[insert\\_or](#)

**Examples**

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

library(oddsratio)
plot_gam(fit_gam, pred = "x2", title = "Predictor 'x2'")
```



# Index

gam: :gam, 4  
glm, 6  
glmmPQL, 6

insert\_or, 2, 5, 8

mgcv: :gam, 4  
mgcv: :gamm, 4

or\_gam, 2–4, 4, 6, 8  
or\_glm, 5, 6

plot\_gam, 2, 3, 5, 7, 8

seq, 8