

Package: gamlss.ggplots (via r-universe)

October 7, 2026

Title Plotting Functions for Generalized Additive Model for Location Scale and Shape

Version 2.1-34

Date 2026-04-25

Description Functions for plotting Generalized Additive Models for Location Scale and Shape from the 'gamlss' package, Stasinopoulos and Rigby (2007) <[doi:10.18637/jss.v023.i07](https://doi.org/10.18637/jss.v023.i07)>, using the graphical methods from 'ggplot2'.

License GPL-2 | GPL-3

URL <https://www.gamlss.com/>

BugReports <https://github.com/gamlss-dev/gamlss.ggplots/issues>

Depends R (>= 4.1.0), gamlss.dist, gamlss (>= 4.3.3), gamlss.foreach

Imports methods, ggridges, ellipse, gamlss.inf, foreach, mgcv, ggplot2, yaImpute, gamlss2, MASS, plotly

Suggests glmnet, reshape2, igraph, networkD3, grid, gridExtra

Additional_repositories <https://gamlss-dev.R-universe.dev>

LazyLoad yes

Config/pak/sysreqs cmake make libicu-dev libuv1-dev libssl-dev

Repository <https://gamlss-dev.r-universe.dev>

Date/Publication 2026-09-20 16:48:14 UTC

RemoteUrl <https://github.com/gamlss-dev/gamlss.ggplots>

RemoteRef HEAD

RemoteSha 0590074cdc967ddc16eef3552cd2824f608fa86a

Contents

gamlss.ggplots-package	2
bivar_fun	4
boot_coef	6

centile_bucket	7
family_pdf	10
fit_PB	13
fitted_cdf	14
fitted_centiles	18
fitted_devianceIncr	21
fitted_leverage	23
fitted_terms	24
histSmo_plot	27
model_GAIC	28
model_pca	31
moment_bucket	32
moment_gray_half	34
pcr_coef_path	36
pe_param	37
pe_pdf	41
prof_term	43
resid_density	45
resid_dtop	46
resid_index	49
resid_qqplot	51
resid_symmetry	53
resid_wp	54
resp_mu	56
xy_median	58
xy_scatter	59
y_hist	63

Index	67
--------------	-----------

gamlss.ggplots-package

Plotting Functions for Generalized Additive Model for Location Scale and Shape

Description

Functions for plotting Generalized Additive Models for Location Scale and Shape from the 'gamlss' package, Stasinopoulos and Rigby (2007) <doi:10.18637/jss.v023.i07>, using the graphical methods from 'ggplot2'.

Details

The DESCRIPTION file: This package was not yet installed at build time.

Index: This package was not yet installed at build time.

The following convention has been used to name the functions:

fitted_NAME: plots concerning fitted values from a single fitted model

resid_NAME: plots concerning residuals from a single fitted model

predict_NAME: plots concerning prediction values from a single fitted model usually having newdata option.

model_NAME: plots concerning different fitted models

where NAME refer to different characteristics.

Author(s)

Mikis Stasinopoulos [aut, cre, cph] (ORCID: <<https://orcid.org/0000-0003-2407-5704>>), Robert Rigby [aut] (ORCID: <<https://orcid.org/0000-0003-3853-1707>>), Fernanda De Bastiani [aut] (ORCID: <<https://orcid.org/0000-0001-8532-639X>>), Julian Merder [ctb]

Maintainer: Mikis Stasinopoulos <d.stasinopoulos@gre.ac.uk>

References

- Rigby, R. A. and Stasinopoulos D. M. (2005). Generalized additive models for location, scale and shape,(with discussion), *Appl. Statist.*, **54**, part 3, pp 507-554.
- Rigby, R. A., Stasinopoulos, D. M., Heller, G. Z., and De Bastiani, F. (2019) *Distributions for modeling location, scale, and shape: Using GAMLSS in R*, Chapman and Hall/CRC, doi:10.1201/9780429298547. An older version can be found in <https://www.gamlss.com/>.
- Stasinopoulos D. M. Rigby R.A. (2007) Generalized additive models for location scale and shape (GAMLSS) in R. *Journal of Statistical Software*, Vol. **23**, Issue 7, Dec 2007, doi:10.18637/jss.v023.i07.
- Stasinopoulos D. M., Rigby R.A., Heller G., Voudouris V., and De Bastiani F., (2017) *Flexible Regression and Smoothing: Using GAMLSS in R*, Chapman and Hall/CRC. doi:10.1201/b21973
- Stasinopoulos, M. D., Rigby, R. A., and De Bastiani F., (2018) GAMLSS: a distributional regression approach, *Statistical Modelling*, Vol. **18**, pp, 248-273, SAGE Publications Sage India: New Delhi, India.
- Stasinopoulos, M.D., Kneib, T., Klein, N., Mayr, A. and Heller, G.Z., (2024). *Generalized Additive Models for Location, Scale and Shape: A Distributional Regression Approach, with Applications* (Vol. **56**). Cambridge University Press.
- (see also <https://www.gamlss.com/>).

See Also

[gamlss](#), [gamlss.family](#)

Examples

```
library(gamlss)
m1 <- gamlss(y~pb(x), data=abdom)
resid_index(m1)
```

bivar_fun

*Function to Plot Two Dimensinal Functions***Description**

The function here plot two dimensional functions. The first two functions are using ‘ggplot2.’

The function `bivar_fun()` is a generic function which can plot the contour plot of any two demen-sional function $f(x,y)$.

The function `bivar_pdf()` is a similar function to `bivar_fun()`, in that it can plot can plot any two dimensional pdf, $f(x,y)$, with the additinal feature that it plots also the HPR (hight probability region) as one of the contour plots.

Usage

```
bivar_fun(fun, xlim = c(-3, 3), ylim = c(-3, 3),
          n = 100, xlab = "x", ylab = "y", ...)

bivar_pdf(pdf, xlim = c(-3, 3), ylim = c(-3, 3),
          n = 200, xlab = "x", ylab = "y", HDV = TRUE, ...)

bivar_fun_plotly(f, xlim = c(-3, 3), ylim = c(-3, 3),
                 n = 100, ...)

bivar.fun.persp(f, xlim = c(-3, 3), ylim = c(-3, 3),
                n = 100, colour = "lightblue", theta = 30, phi = 30,
                xlab = "x", ylab = "y", zlab = "f(x,y)",
                main = "f(x,y)", ...)

bivar.fun.image(f, xlim = c(-3, 3), ylim = c(-3, 3),
                n = 100, xlab = "x", ylab = "y", main = "f()",
                colour = hcl.colors(20), contour = TRUE, ...)

bivar.fun.contour(f, xlim = c(-3, 3), ylim = c(-3, 3),
                  n = 100, xlab = "x", ylab = "y", ...)

bivar_fun_plotly(f, xlim = c(-3, 3), ylim = c(-3, 3),
                 n = 100, ...)
```

Arguments

<code>fun, f</code>	any two dimensional function $f(x,y)$
<code>pdf</code>	any two dimensinal joint pdf
<code>xlim</code>	the limits of the x
<code>ylim</code>	the limits of the x

n	the number of grid
xlab	the x lable
ylab	the y label
zlab	the z label
HDV	whether to show the hight density values (50%)
colour	the aproprate colour
contour	whether to plot the contour
theta, phi	angles defibne the view direction
main	the title
...	additinal argument passes to the relevent function

Value

The results is a ggplot2 function.

Author(s)

Mikis Stasinopoulos

References

Rigby, R. A., Stasinopoulos, D. M., Heller, G. Z., and De Bastiani, F. (2019) *Distributions for modeling location, scale, and shape: Using GAMLSS in R*, Chapman and Hall/CRC. An older version can be found in <https://www.gamlss.com/>.

Stasinopoulos D. M. Rigby R.A. (2007) Generalized additive models for location scale and shape (GAMLSS) in R. *Journal of Statistical Software*, Vol. **23**, Issue 7, Dec 2007, <https://www.jstatsoft.org/v23/i07/>.

Stasinopoulos D. M., Rigby R.A., Heller G., Voudouris V., and De Bastiani F., (2017) *Flexible Regression and Smoothing: Using GAMLSS in R*, Chapman and Hall/CRC.

Stasinopoulos, M.D., Kneib, T., Klein, N., Mayr, A. and Heller, G.Z., (2024). *Generalized Additive Models for Location, Scale and Shape: A Distributional Regression Approach, with Applications* (Vol. **56**). Cambridge University Press.

(see also <https://www.gamlss.com/>).

See Also

xy_scatter

Examples

```
f_mix <- function(x, y)
{
  0.5 * exp(-(x^2 + y^2)) +
  0.5 * exp(-((x-2)^2 + (y-2)^2))
}

bivar_pdf(f_mix, xlim = c(-3, 5), ylim = c(-3, 5))
bivar_fun(f_mix, xlim = c(-3, 5), ylim = c(-3, 5))
```

boot_coef

*Plotting Bootstrap Coefficients***Description**

The function `boot_coef()` plots in one or multiple pages the results from a bootstrap generated by the function `BayesianBoot()`, `NonParametricBoot` or `nonpar_boot()`.

The function `boot_coef_one()` plots a single parameter.

Usage

```
boot_coef(x, terms = NULL, hist.col = "black",
          hist.fill = "white", dens.fill = "#FF6666",
          alpha = 0.2, nrow = NULL, ncol = NULL,
          plots.per.page = 9, one.by.one = FALSE, title, ...)

boot_coef_one(x, par = 1, rug = TRUE, alpha = 0.2, hist.col = "black",
              hist.fill = "white", line.col = "gray",
              dens.fill = "#FF6666", title, ...)
```

Arguments

<code>x</code>	a <code>Bayesian.boot</code> or <code>NonParametric.boot</code> object
<code>terms</code>	which terms to plot (default <code>NULL</code> means all terms)
<code>par</code>	which parameter to plot
<code>hist.col</code>	colour of the border histogram
<code>hist.fill</code>	the colour of the histogram
<code>dens.fill</code>	the colour of the density estimate
<code>alpha</code>	transparency constant
<code>nrow</code>	how many rows
<code>ncol</code>	how many columns
<code>plots.per.page</code>	the maximum plots per page
<code>one.by.one</code>	whether single plots
<code>rug</code>	whether rug is required for <code>boot_coef_one()</code>
<code>line.col</code>	the vertical line colour for <code>boot_coef_one()</code>
<code>title</code>	the title
<code>...</code>	for more argument

Details

The function plots in one or multiple pages the results from a bootstrap simulation

Value

Creates a ggplot object

Author(s)

Mikis Stasinopoulos

References

Rigby, R. A. and Stasinopoulos D. M. (2005). Generalized additive models for location, scale and shape,(with discussion), *Appl. Statist.*, **54**, part 3, pp 507-554.

Rigby, R. A., Stasinopoulos, D. M., Heller, G. Z., and De Bastiani, F. (2019) *Distributions for modeling location, scale, and shape: Using GAMLSS in R*, Chapman and Hall/CRC. An older version can be found in <https://www.gamlss.com/>.

Stasinopoulos D. M. Rigby R.A. (2007) Generalized additive models for location scale and shape (GAMLSS) in R. *Journal of Statistical Software*, Vol. **23**, Issue 7, Dec 2007, <https://www.jstatsoft.org/v23/i07/>.

Stasinopoulos D. M., Rigby R.A., Heller G., Voudouris V., and De Bastiani F., (2017) *Flexible Regression and Smoothing: Using GAMLSS in R*, Chapman and Hall/CRC.

Stasinopoulos, M.D., Kneib, T., Klein, N., Mayr, A. and Heller, G.Z., (2024). *Generalized Additive Models for Location, Scale and Shape: A Distributional Regression Approach, with Applications* (Vol. **56**). Cambridge University Press.

(see also <https://www.gamlss.com/>).

See Also

[term.plot](#)

Examples

```
data(aids)
a <- gamlss(y ~ pb(x) + qrt, data = aids, family = NBI)
registerDoParallel(cores = 2)
B1 <- BayesianBoot(a, B=100)
stopImplicitCluster()
boot_coef(B1)
```

Description

A bucket plot is a graphical way to check the skewness and kurtosis of a continuous variable or the residuals of a fitted GAMLSS model. It plots the centile skewness (tail or central) and transformed centile kurtosis of the variable (or residuals) together with a cloud of points obtained using a non-parametric bootstrap from the original variable (or residuals). It also provides a graphical way of performing a Monte Carlo simulation test on whether the centile skewness and transformed centile kurtosis of the variable of interest are simultaneously equal to zero.

There are two function here:

i) `cenlile_bucket()` for a single bucket plot. Note that `model_cent_bucket()` and `centile_bucket()` are synonymous.

ii) `centile_bucket_wrap()` for multiple bucket plots cut according to terms in the model.

Usage

```
centile_bucket(x, ..., type = c("tail", "central"), weights = NULL,
  no_bootstrap = 99, col_bootstrap = hcl.colors(length.obj,
  palette = "Set 2"), alpha_bootstrap = 1, text_to_show = NULL,
  cex_text = 5, col_text = "black", colour_bucket = FALSE,
  line_width = 0.5, sim_test = FALSE, no_sim_test = 1000,
  col_sim_test = gray(0.7), alpha_sim_test = 0.1, seed_test = 1234)
```

```
model_cent_bucket(x, ..., type = c("tail", "central"), weights = NULL,
  no_bootstrap = 99, col_bootstrap = hcl.colors(length.obj,
  palette = "Set 2"), alpha_bootstrap = 1, text_to_show = NULL,
  cex_text = 5, col_text = "black", colour_bucket = FALSE,
  line_width = 0.5, sim_test = FALSE, no_sim_test = 1000,
  col_sim_test = gray(0.7), alpha_sim_test = 0.1, seed_test = 1234)
```

```
centile_bucket_wrap(x, ..., type = c("tail", "central"), weights = NULL,
  xvar = NULL, n_inter = 4, no_bootstrap = 99,
  col_bootstrap = hcl.colors(length.obj, palette = "Set 2"),
  alpha_bootstrap = 1, text_to_show = NULL, check_overlap_text = FALSE,
  cex_text = 5, col_text = "black", colour_bucket = FALSE,
  line_width = 0.5, sim_test = FALSE, no_sim_test = 1000,
  col_sim_test = gray(0.7), alpha_sim_test = 0.1, seed_test = 1234)
```

Arguments

<code>x</code>	<code>x</code> should be a continuous vector of a GAMLSS fitted model.
<code>...</code>	for more that one continuous vectors or fitted models
<code>type</code>	whether "tail" of "central" skewness and kurtosis
<code>weights</code>	if priors weights are needed
<code>no_bootstrap</code>	the number of bootstrap samples for the cloud around the point of skewness and kurtosis.

col_bootstrap	The colour of the bootstrap samples
alpha_bootstrap	The transparency parameter of the bootstrap samples.
text_to_show	what text to show in the plots, default the names of vectors or models
cex_text	the character size of the text
col_text	the colour of the text
colour_bucket	whether colour or gray lines in the bucket
line_width	the line width
sim_test	whether to Monte Carlo simulation is needed to check the null hypothesis that there is no centile skewness and transformed centile kurtosis in the sample.
no_sim_test	The number of simulation for the test
col_sim_test	the colour used for displaying the Monte Carlo test values
alpha_sim_test	The transparency parameter of the Monte Carlo samples.
seed_test	A seed value for the Monte Carlo simulation.
xvar	the x term
n_inter	how many intervals needed
check_overlap_text	whether to check overlapping text

Details

More details about centile bucket plots is given in De Bastiani *et al.* (2022)

Value

A plot displaying the centile skewness and transformed centile kurtosis of the sample or residual of a model.

Note

The bucket plot provides an additional residual diagnostic tool that can be used for fitted model checking, alongside other diagnostic tools, for example worm plots, and Q (and Z) statistics.

Author(s)

Mikis Stasinopoulos, Bob Rigby and Fernanda De Bastiani

References

- De Bastiani, F., Stasinopoulos, D. M., Rigby, R. A., Heller, G. Z., and Lucas A. (2022) Bucket Plot: A Visual Tool for Skewness and Kurtosis Comparisons. To be published.
- Rigby, R. A. and Stasinopoulos D. M. (2005). Generalized additive models for location, scale and shape,(with discussion), *Appl. Statist.*, **54**, part 3, pp 507-554.

Rigby, R. A., Stasinopoulos, D. M., Heller, G. Z., and De Bastiani, F. (2019) *Distributions for modeling location, scale, and shape: Using GAMLSS in R*, Chapman and Hall/CRC. doi:10.1201/9780429298547 An older version can be found in <https://www.gamlss.com/>.

Stasinopoulos D. M. Rigby R.A. (2007) Generalized additive models for location scale and shape (GAMLSS) in R. *Journal of Statistical Software*, Vol. **23**, Issue 7, Dec 2007, doi:10.18637/jss.v023.i07.

Stasinopoulos D. M., Rigby R.A., Heller G., Voudouris V., and De Bastiani F., (2017) *Flexible Regression and Smoothing: Using GAMLSS in R*, Chapman and Hall/CRC. doi:10.1201/b21973

Stasinopoulos, M. D., Rigby, R. A., and De Bastiani F., (2018) GAMLSS: a distributional regression approach, *Statistical Modelling*, Vol. **18**, pp, 248-273, SAGE Publications Sage India: New Delhi, India.

Stasinopoulos, M.D., Kneib, T., Klein, N., Mayr, A. and Heller, G.Z., (2024). *Generalized Additive Models for Location, Scale and Shape: A Distributional Regression Approach, with Applications* (Vol. **56**). Cambridge University Press.

(see also <https://www.gamlss.com/>).

See Also

[wp](#), [Q.stats](#)

Examples

```
m1 <- gamlss(R~pb(F1)+pb(A), data=rent, family=GA)
centile_bucket(m1)

centile_bucket_wrap(m1, xvar=rent$A)
```

family_pdf

Plotting Probabilities Density Functions (pdf's) for GAMLSS

Description

The function `family_pdf()` takes a GAMLSS family distribution and plots different pdf's according to the specified parameters.

The function `fitted_pdf()` takes a `gamlss` fitted object and plots the fitted distributions for specified observations.

The function `fitted_pdf_data()` it does the same as `fitted_pdf()` but it adds also the observation values as grey vertical lines.

The function `predict_pdf()` takes a fitted object and test data and plots the predictive pdf's.

Usage

```
family_pdf(family = NO(), mu = NULL, sigma = NULL, nu = NULL, tau = NULL,
           title, from = 0, to = 10, no.points = 201,
           alpha = 0.4, col.fill = hcl.colors(lobs, palette = "viridis"),
           size.segment = 1.5, plot.point = TRUE, size.point = 1,
           plot.line = TRUE, size.line = 0.2, ...)

fitted_pdf(model, obs, title, from = 0, to = 10, no.points = 201, alpha = 0.4,
           col.fill = hcl.colors(lobs, palette = "viridis"),
           size.segment = 1.5, plot.point = TRUE, size.point = 1, plot.line = TRUE,
           size.line = 0.2, ...)

fitted_pdf_data(model, obs, from, to,
                colour = hcl.colors(length(obs), palette = "viridis"), ...)

predict_pdf(model, newdata, title, from = 0, to = 10, no.points = 201,
           alpha = 0.4, col.fill = hcl.colors(lobs, palette = "viridis"),
           size.segment = 1.5, plot.point = TRUE, size.point = 1,
           plot.line = TRUE, size.line = 0.2, ...)
```

Arguments

family	A GAMLSS family
model	A GAMLSS fitted model
obs	observations to plot fitted distributions
newdata	for test data
mu	the mu parameter value(s)
sigma	the sigma parameter value(s)
nu	the nu parameter value(s)
tau	the tau parameter value(s)
title	a different title for the default
from	minimum value for the response
to	maximum value for the response
no.points	number of points (relevant for continuous responses)
alpha	transparency factor
col.fill	the colour pallet default is <code>hcl.colors(lobs, palette="viridis")</code>
colour	the colour pallet default is <code>hcl.colors(lobs, palette="viridis")</code>
size.segment	for discrete responses the size of the bars
plot.point	for discrete responses whether to put poits on the top of the bars
size.point	for discrete responses
plot.line	for discrete responses whether to joint the bars with lines
size.line	for discrete responses the size of the joining lines
...	for extra argumnets

Details

The functions `family_pdf()` and `fitted_pdf()` are ggplot version of the function `pdf.plot()` used to plot fitted distributions of GAMLSS family at specified observation values. Note that the range of the response has to be specified using the argument `from` to `to`.

For discrete fitted distributions maybe increase the value of `alpha` for clearer plot.

For binomial type of data (discrete response with upper limit) the function `family_pdf()` takes the argument `to` as the binomial denominator, For fitted model with binomial type responses the function `fitted_pdf()` takes the binomial denominator from the fitted model and set the argument `to` to the maximum of those binomial denominators.

Value

Creates a plot

Author(s)

Mikis Stasinopoulos, Bob Rigby and Fernanda De Bastiani

References

Rigby, R. A., Stasinopoulos, D. M., Heller, G. Z., and De Bastiani, F. (2019) *Distributions for modeling location, scale, and shape: Using GAMLSS in R*, Chapman and Hall/CRC. An older version can be found in <https://www.gamlss.com/>.

Stasinopoulos D. M. Rigby R.A. (2007) Generalized additive models for location scale and shape (GAMLSS) in R. *Journal of Statistical Software*, Vol. **23**, Issue 7, Dec 2007, <https://www.jstatsoft.org/v23/i07/>.

Stasinopoulos D. M., Rigby R.A., Heller G., Voudouris V., and De Bastiani F., (2017) *Flexible Regression and Smoothing: Using GAMLSS in R*, Chapman and Hall/CRC.

Stasinopoulos, M.D., Kneib, T., Klein, N., Mayr, A. and Heller, G.Z., (2024). *Generalized Additive Models for Location, Scale and Shape: A Distributional Regression Approach, with Applications* (Vol. **56**). Cambridge University Press.

(see also <https://www.gamlss.com/>).

See Also

[gamlss](#), [resid_density](#)

Examples

```
#####
# function fitted_pdf
# continuous variabe
a1 <- gamlss(y~pb(x),sigma.fo=~pb(x), data=abdom, family=L0)
fitted_pdf(a1, obs=c(10,15,20), from=30, to=100)
# count data
p1 <- gamlss(y~pb(x)+qrt, data=aids, family=NBI)
fitted_pdf(p1, obs=c(10:15), from=25, to=130, alpha=.9)
# binomial type
```

```

h<-gamlss(y~ward+loglos+year, sigma.formula=~year+ward, family=BB, data=aep)
fitted_pdf(h, obs=c(10:15), alpha=.9)
#####
# function predict_pdf
predict_pdf(a1, newdata=abdom[10:20, ], from=30, to=100)
# count data
predict_pdf(p1, newdata=aids[10:15, ], from=30, to=150)
# binomial
predict_pdf(h, newdata=aep[10:15, ], from=0, to=20)
#####
# function family_pdf
# continuous
family_pdf(from=-5,to=5, mu=0, sigma=c(.5,1,2))
# count data
family_pdf(NBI, to=15, mu=1, sigma=c(.5,1,2), alpha=.9, size.segment = 3)
# binomial type
family_pdf(BB, to=15, mu=.5, sigma=c(.5,1,2), alpha=.9, , size.segment = 3)

```

fit_PB

P-spline smoother

Description

The function `fit_PB()` fits a P-spline univariate smoother [Eilers and Marx (1996)] to y against the x with prior weights weights.

Usage

```

fit_PB(x, y, weights, data, xmin, xmax, nseg = 20,
       lambda = 10, order = 2, degree = 3, max.df = 20,
       ylim, plot = FALSE, col.ribbon = "pink")

```

Arguments

<code>x</code>	the explanatory variable
<code>y</code>	the response
<code>weights</code>	possible prior weights (set to one by default)
<code>data</code>	the data frame where x , y and <code>weights</code> are coming from
<code>xmin</code>	the x minimum if different from <code>min(x)</code>
<code>xmax</code>	the x maximum if different from <code>max(x)</code>
<code>nseg</code>	the number of knots
<code>lambda</code>	the smotthing parameter
<code>order</code>	the ordr of the difference
<code>degree</code>	the degree of the piewise polynonmial
<code>max.df</code>	the maximum allowed degress of freedom
<code>ylim</code>	the <code>ylim</code> in the plot
<code>plot</code>	whether to plot the results
<code>col.ribbon</code>	the color in the se of the fitted values

Value

A object `Psplines` is produced with methods `print()`, `coef()`, `deviance()`, `fitted()`, `predict()` and `resid()`.

Note

The function `fit_PB()` is an engine for getting the maximal correlation between two continuous variables. It can be also used on its own as a smoother.

Author(s)

Mikis Stasinopoulos

References

Eilers, P. H. C. and Marx, B. D. (1996). Flexible smoothing with B-splines and penalties (with comments and rejoinder). *Statist. Sci.*, **11**, 89-121.

Rigby, R. A., Stasinopoulos, D. M., Heller, G. Z., and De Bastiani, F. (2019) *Distributions for modeling location, scale, and shape: Using GAMLSS in R*, Chapman and Hall/CRC. doi:10.1201/9780429298547.

Stasinopoulos D. M., Rigby R.A., Heller G., Voudouris V., and De Bastiani F., (2017) *Flexible Regression and Smoothing: Using GAMLSS in R*, Chapman and Hall/CRC. doi:10.1201/b21973

Stasinopoulos, M.D., Kneib, T., Klein, N., Mayr, A. and Heller, G.Z., (2024). *Generalized Additive Models for Location, Scale and Shape: A Distributional Regression Approach, with Applications* (Vol. 56). Cambridge University Press.

(see also <https://www.gamlss.com/>).

See Also

ACE

Examples

```
data(abdom)
m1 <- fit_PB(x,y, data=abdom)
```

Description

The function `family_cdf()` takes a GAMLSS family distribution and plots different pdf's according to the specified parameters.

The function `fitted_cdf()` takes a gamlss fitted object and plots the fitted distributions for specified observations.

The function `fitted_survival()` takes a gamlss fitted object and plots the the survival function for specified observations.

The function `fitted_cdf_data()` is similar to `fitted_cdf()` but also adds the data points as vertical lines.

The function `fitted_survival_data()` is similar to `fitted_survival()` but also adds the data points as vertical lines.

The function `predict_pdf()` (NOT IMPLEMENTED YET) takes a fitted object and test data and plots the predictive cdf's.

Usage

```
fitted_cdf(model, obs, title, from = 0, to = 10, no.points = 201,
           alpha = 1, size.line = 1.2,
           col.fill = hcl.colors(lobs, palette = "viridis"),
           size.segment = 1.5, size.point = 1,
           plot.line = TRUE, size.line.disc = 0.2, lower.tail = TRUE, ...)
```

```
fitted_survival(model, obs, title, from = 0, to = 10, no.points = 201,
                alpha = 1, size.line = 1.2,
                col.fill = hcl.colors(lobs, palette = "viridis"),
                size.segment = 1.5, size.point = 1,
                plot.line = TRUE, size.line.disc = 0.2, lower.tail = TRUE, ...)
```

```
fitted_cdf_data(model, obs, from, to,
                 colour = hcl.colors(length(obs), palette = "viridis"), ...)
```

```
fitted_survival_data(model, obs, from, to,
                     colour = hcl.colors(length(obs), palette = "viridis"), ...)
```

```
predict_cdf(model, newdata, title, from = 0, to = 10, no.points = 201,
            alpha = 0.4, size.line = 1.2,
            col.fill = hcl.colors(lobs, palette = "viridis"),
            size.segment = 1.5, plot.point = TRUE, size.point = 1,
            plot.line = TRUE, size.line.disc = 0.2, lower.tail = TRUE, ...)
```

```
family_cdf(family = NO(), mu = NULL, sigma = NULL, nu = NULL,
           tau = NULL, title, from = 0, to = 10, no.points = 201,
           alpha = 0.4, size.line = 1.2, col.fill = hcl.colors(lobs,
           palette = "viridis"), size.segment = 1.5, plot.point = TRUE,
           size.point = 1, plot.line = TRUE, lower.tail = TRUE, ...)
```

Arguments

family	A GAMLSS family
model	A GAMLSS fitted model
obs	observations to plot fitted distributions
newdata	for test data
mu	the mu parameter value(s)
sigma	the sigma parameter value(s)
nu	the nu parameter value(s)
tau	the tau parameter value(s)
title	a different title for the default
from	minimum value for the response
to	maximum value for the response
no.points	number of points (relevant for continuous responses)
alpha	transparency factor
col.fill	the colour pallet default is <code>hcl.colors(lobs, palette="viridis")</code>
colour	the colour pallet default is <code>hcl.colors(lobs, palette="viridis")</code>
size.segment	for discrete responses the size of the bars
plot.point	for discrete responses whether to put poits on the top of the bars
size.point	for discrete responses
plot.line	for discrete responses whether to joint the bars with lines
size.line.disc	for discrete responses the size of the joining lines
size.line	The size of the lines
lower.tail	if TRUE cdf is plotted if FALSE the survival function
...	for extra argumnets

Details

The functions `family_cdf()`, `fitted_cdf()`, and `predict_cdf()` are function to plot cdf's for a `gamlss.family`, fitted `gamlss` model or predictive `gamlss` model, respectively.

For discrete fitted distributions maybe increase the value of `alpha` for clearer plot.

For binomial type of data (discrete response with upper limit) the function `family_cdf()` takes the argument `to` as the binomial denominator, For fitted model with binomial type responses the function `fitted_cdf()` takes the binomial denominator form the fitted model and set the argument `to` to the maximum of those binomial denominators.

Value

Creates a plot

Author(s)

Mikis Stasinopoulos, Bob Rigby and Fernanda De Bastiani

References

Rigby, R. A., Stasinopoulos, D. M., Heller, G. Z., and De Bastiani, F. (2019) *Distributions for modeling location, scale, and shape: Using GAMLSS in R*, Chapman and Hall/CRC. An older version can be found in <https://www.gamlss.com/>.

Stasinopoulos D. M. Rigby R.A. (2007) Generalized additive models for location scale and shape (GAMLSS) in R. *Journal of Statistical Software*, Vol. **23**, Issue 7, Dec 2007, <https://www.jstatsoft.org/v23/i07/>.

Stasinopoulos D. M., Rigby R.A., Heller G., Voudouris V., and De Bastiani F., (2017) *Flexible Regression and Smoothing: Using GAMLSS in R*, Chapman and Hall/CRC.

Stasinopoulos, M.D., Kneib, T., Klein, N., Mayr, A. and Heller, G.Z., (2024). *Generalized Additive Models for Location, Scale and Shape: A Distributional Regression Approach, with Applications* (Vol. **56**). Cambridge University Press.

(see also <https://www.gamlss.com/>).

See Also

[gamlss](#)

Examples

```
# function fitted_cdf
# continuous variabe
a1 <- gamlss(y~pb(x),sigma.fo=~pb(x), data=abdom, family=L0)
fitted_cdf(a1, obs=c(10,15,20), from=30, to=100)
fitted_cdf(a1, obs=c(10,15,20), from=30, to=100, lower.tail=FALSE)
# count data
p1 <- gamlss(y~pb(x)+qrt, data=aids, family=NBI)
fitted_cdf(p1, obs=c(10:15), from=10, to=130, alpha=.9)
fitted_cdf(p1, obs=c(10:15), from=10, to=130, alpha=.9, lower.tail=FALSE)
# binomial type
h<-gamlss(y~ward+loglos+year, sigma.formula=~year+ward, family=BB, data=aep)
fitted_cdf(h, obs=c(10:15), alpha=.9)
fitted_cdf(h, obs=c(10:15), alpha=.9, lower.tail=FALSE)
#####
# function predict_pdf
predict_cdf(a1, newdata=abdom[c(10,15,20), ], from=30, to=100)
predict_cdf(a1, newdata=abdom[10:20, ], from=30, to=100, lower.tail=FALSE)
# count data
predict_cdf(p1, newdata=aids[10:15, ], from=10, to=150)
predict_cdf(p1, newdata=aids[10:15, ], from=10, to=150, lower.tail=FALSE)
# binomial
predict_cdf(h, newdata=aep[10:15, ], from=0, to=20)
predict_cdf(h, newdata=aep[10:15, ], from=0, to=20, lower.tail=FALSE)
#####
# function family_cdf
# continuous
family_cdf(from=-5,to=5, mu=0, sigma=c(.5,1,2))
family_cdf(from=-5,to=5, mu=0, sigma=c(.5,1,2), lower.tail=FALSE)
# count data
family_cdf(NBI, to=15, mu=1, sigma=c(.5,1,2), alpha=.9, size.segment = 3)
```

```
family_cdf(NBI, to=15, mu=1, sigma=c(.5,1,2), alpha=.9, size.segment = 3, lower.tail=FALSE)
# binomial type
family_cdf(BB, to=15, mu=.5, sigma=c(.5,1,2), alpha=.9, , size.segment = 3)
family_cdf(BB, to=15, mu=.5, sigma=c(.5,1,2), alpha=.9, , size.segment = 3, lower.tail=FALSE)
```

fitted_centiles

Plotting centile (growth) curves

Description

The function `fitted_centiles()` plots centiles curves for distributions belonging to the GAMLSS family of distributions. The plot is equivalent to the standard plot of `gamlss:::centiles()` without a legend.

The function `fitted_centiles_legend()` plots centiles curves for distributions belonging to the GAMLSS family of distributions and it is equivalent to the standard plot of `gamlss:::centiles()` with a legend. The function is slower than `fitted_centiles()` since in order to plot the legend the data have to be expanded.

The function `fitted_centiles_gap()` is based on the function `fitted_centiles_legend()` but it can cope with data that have gaps in the x-variable. A typical example is data collected during pregnancy when some weeks do not have data, see the data APL in the package `gamlss.data`.

The function `model_centiles()` plots centile curves for more than one model. There is no equivalent plot in the original GAMLSS centile plots but it performs the same function as `gamlss:::centiles.com()` which compares centiles from different models.

Usage

```
fitted_centiles(obj, xvar,
               cent = c(99.4, 98, 90, 75, 50, 25, 10, 2, 0.4),
               points = TRUE, point.col = "gray",
               point.size = 1, line.size = 0.8,
               line.col = hcl.colors(10, palette = "Dark 2"),
               line.type = rep(1, length(cent)),
               xlab = NULL, ylab = NULL, ylim, title, ...)

fitted_centiles_legend(obj, xvar,
                     cent = c(99.4, 98, 90, 75, 50, 25, 10, 2, 0.4),
                     points = TRUE, point.col = "gray", point.size = 1,
                     line.size = 0.8, line.col = hcl.colors(ncent,
                     palette = "Dark 2"), line.type = rep(1, length(cent)),
                     show.legend = TRUE, save.data = FALSE, title,
                     xlab = NULL, ylab = NULL, ylim, ...)

fitted_centiles_gap(obj, xvar,
                  cent = c(98, 90, 75, 50, 25, 10, 2),
                  points = TRUE, point.col = "gray", point.size = 1,
                  line.size = 0.8, line.col = hcl.colors(ncent,
```

```

palette = "Dark 2"), line.type = rep(1, length(cent)),
show.legend = TRUE, save.data = FALSE, title,
xlab = NULL, ylab = NULL, ylim, ...)

model_centiles(obj, ..., cent = c(97, 90, 75, 50, 25, 10, 3),
xvar, xlab = "age", points = TRUE,
point.col = gray(0.8),
point.size = 0.05, line.size = 0.7,
line.col = hcl.colors(ncent,palette = "Dark 2"),
ncol = 2, nrow = ceiling(nnames/ncol), in.one = FALSE,
title)

```

Arguments

obj	a fitted gamlss object
xvar	the (unique) explanatory variable
cent	a vector with elements the % centile values for which the centile curves have to be evaluated (note that the order is from the highest to the lowest so legend and the plots are maching)
points	whether to plot the points (TRUE) of the data or not (FALSE)
point.col	the colour of the points
point.size	the zize of the points
line.size	the sized of the centile lines
line.col	the colour of the centile lines
line.type	the type of line (different types of lines for each centile are working with fitted_centiles_legend)
xlab	the label of the x-axis variable
ylab	the label of the resposnse variable
ylim	the limits of the resposnse variable
in.one	whether the model_centile plot should be one or multiple
title	the title if need it otherwise a dfault title is pronted
show.legend	whether to show the legend
save.data	whether to save the data.frame of the plot
nrow	the number of rows in the model_centiles() plot
ncol	the number of columns in the model_centiles() plot
...	for extra arguments for fitted_centiles(), and fitted_centiles.legend() and extra models for model_centiles()

Details

Centiles are calculated using the fitted values in obj and xvar must correspond exactly to the predictor in obj to plot correctly.

Value

A plot is created

Warning

This function is appropriate only when one continuous explanatory variable is fitted in the model

Author(s)

Mikis Stasinopoulos, Bob Rigby and Fernanda de Bastiani

References

Rigby, R. A. and Stasinopoulos D. M. (2005). Generalized additive models for location, scale and shape,(with discussion), *Appl. Statist.*, **54**, part 3, pp 507-554.

Rigby, R. A., Stasinopoulos, D. M., Heller, G. Z., and De Bastiani, F. (2019) *Distributions for modeling location, scale, and shape: Using GAMLSS in R*, Chapman and Hall/CRC. An older version can be found in <https://www.gamlss.com/>.

Stasinopoulos D. M. Rigby R.A. (2007) Generalized additive models for location scale and shape (GAMLSS) in R. *Journal of Statistical Software*, Vol. **23**, Issue 7, Dec 2007, <https://www.jstatsoft.org/v23/i07/>.

Stasinopoulos D. M., Rigby R.A., Heller G., Voudouris V., and De Bastiani F., (2017) *Flexible Regression and Smoothing: Using GAMLSS in R*, Chapman and Hall/CRC.

Stasinopoulos, M.D., Kneib, T., Klein, N., Mayr, A. and Heller, G.Z., (2024). *Generalized Additive Models for Location, Scale and Shape: A Distributional Regression Approach, with Applications* (Vol. **56**). Cambridge University Press.

(see also <https://www.gamlss.com/>).

See Also

[centiles](#)

Examples

```
data(abdom)
h<-gamlss(y~pb(x), sigma.formula=~pb(x), family=BCTo, data=abdom)
h1 <- gamlss(y~pb(x), sigma.formula=~pb(x), family=L0, data=abdom)
fitted_centiles(h)
fitted_centiles_legend(h)
model_centiles(h, h1)
```

fitted_devianceIncr *Plotting the deviance increment of GAMLSS*

Description

There are two plotting function here:

i) fitted_devianceIncr() plots the fitted model deviance components. This is useful for identifying observations with unusual y-values (given the current fitted distribution).

iii) model_devianceIncr_diff plots the difference of deviance increments from two fitted GAMLSS model. This function is useful if the GAIC and the residuals contradict each other. For example the GAIC is better for model 1 but the residuals look lot better for model 2. This can happens if the two distributions are better suited to fit different parts of the response distribution i.e. one model fits the center better but the other fits the tail better.

Usage

```
fitted_devianceIncr(obj, plot = TRUE, title, quan.val = 0.99,
  annotate = TRUE, line.col = "steelblue4",
  point.col = "darkblue", annot.col = "white",
  newdata = NULL)
```

```
model_devianceIncr_diff(model1, model2, against = "index",
  tol = 20, data, newdata)
```

Arguments

obj	a GAMLSS fitted object
plot	whether to create just the plot or save also the values with high deviance increment
title	a tittle if needed it.
quan.val	The quantile values of the deviance increment from which the obsrevrvation should be identify
annotate	whether to plot the quantile values above in the plot.
line.col	the colour of the line
point.col	the colour of the points
annot.col	the colour of the annotation for the deviance increment plot
model1	The first fitted GAMLSS model
model2	The second fitted GAMLSS model
against	you can plot the deviance increment an index, the response or an x-variable
tol	if the absolute value of deviance increment exceeds the tol the number of the observation is plotted
data	The data if can not be found from model1
newdata	evaluates the function in new data

Details

The functions are diagnostic tools to check unusual observations in the response.

Value

return a plot

Author(s)

Mikis Stasinopoulos, Rober Rigby and Fernanda de Bastiani

References

Rigby, R. A. and Stasinopoulos D. M. (2005). Generalized additive models for location, scale and shape,(with discussion), *Appl. Statist.*, **54**, part 3, pp 507-554.

Rigby, R. A., Stasinopoulos, D. M., Heller, G. Z., and De Bastiani, F. (2019) *Distributions for modeling location, scale, and shape: Using GAMLSS in R*, Chapman and Hall/CRC. An older version can be found in <https://www.gamlss.com/>.

Stasinopoulos D. M. Rigby R.A. (2007) Generalized additive models for location scale and shape (GAMLSS) in R. *Journal of Statistical Software*, Vol. **23**, Issue 7, Dec 2007, <https://www.jstatsoft.org/v23/i07/>.

Stasinopoulos D. M., Rigby R.A., Heller G., Voudouris V., and De Bastiani F., (2017) *Flexible Regression and Smoothing: Using GAMLSS in R*, Chapman and Hall/CRC.

Stasinopoulos, M.D., Kneib, T., Klein, N., Mayr, A. and Heller, G.Z., (2024). *Generalized Additive Models for Location, Scale and Shape: A Distributional Regression Approach, with Applications* (Vol. **56**). Cambridge University Press.

(see also <https://www.gamlss.com/>).

See Also

[gamlss](#)

Examples

```
m1 <- gamlss(R~pb(F1)+pb(A)+H+loc, data=rent, family=GA )
m2 <- gamlss(R~pb(F1)+pb(A)+H+loc, data=rent, family=NO )
fitted_devianceIncr(m1)
model_devianceIncr_diff(m1,m2, against="F1")
```

fitted_leverage	<i>Plot of the linear leverage of a GAMLSS model</i>
-----------------	--

Description

This is plot of the "linear" leverage of a GAMLSS fitted model. By linear we mean the leverage (hat-values) we would have obtain in all the explanatory variables for all distribution parameters where put together and used to fit a linear model to the response. The "linear" leverage is them the hat-values obtained by fitting this simple linear model. Hopefully the "linear" leverage can indicate observations with extreme values in the x's. Note that observations with hight linear leverage may not be influential in the GAMLSS fitting especially if the x-variables are fitted using smoothers.

Usage

```
fitted_leverage(obj, plot = TRUE, title, quan.val = 0.99,  
  annotate = TRUE, line.col = "steelblue4",  
  point.col = "steelblue4", annot.col = "darkred")
```

Arguments

obj	A GAMLSS fitted model
plot	whether to plot ot not
title	for different title than the default
quan.val	which quantile value of the leverage should be taked to indicate the observation values
annotate	whether to annotate the extreme levarages
line.col	the colour of the lines
point.col	the colout of the points
annot.col	the colour used for annotation

Value

Returns a plot of the linear leverage against index.

Author(s)

Mikis Stasinopoulos, Bob Rigby and Fernanda De Bastiani

References

Rigby, R. A. and Stasinopoulos D. M. (2005). Generalized additive models for location, scale and shape,(with discussion), *Appl. Statist.*, **54**, part 3, pp 507-554.

Rigby, R. A., Stasinopoulos, D. M., Heller, G. Z., and De Bastiani, F. (2019) *Distributions for modeling location, scale, and shape: Using GAMLSS in R*, Chapman and Hall/CRC. An older version can be found in <https://www.gamlss.com/>.

Stasinopoulos D. M., Rigby R.A. (2007) Generalized additive models for location scale and shape (GAMLSS) in R. *Journal of Statistical Software*, Vol. **23**, Issue 7, Dec 2007, <https://www.jstatsoft.org/v23/i07/>.

Stasinopoulos D. M., Rigby R.A., Heller G., Voudouris V., and De Bastiani F., (2017) *Flexible Regression and Smoothing: Using GAMLSS in R*, Chapman and Hall/CRC.

Stasinopoulos, M.D., Kneib, T., Klein, N., Mayr, A. and Heller, G.Z., (2024). *Generalized Additive Models for Location, Scale and Shape: A Distributional Regression Approach, with Applications* (Vol. **56**). Cambridge University Press.

(see also <https://www.gamlss.com/>).

See Also

[gamlss](#)

Examples

```
m1 <- gamlss(R~pb(F1)+pb(A)+loc+H, data=rent, family=GA)
fitted_leverage(m1)
```

fitted_terms

Plotting fitted additive terms

Description

The function `fitted_terms()` is doing what the function `term.plot()` is doing for GAMLSS models but it uses `ggplot2` package. The function `pe_terms()` is synonymous to `fitted_terms()` in the package **gamlss**.

Usage

```
fitted_terms(object,
  what = c("mu", "sigma", "nu", "tau"),
  parameter = NULL, data = NULL, terms = NULL,
  envir = environment(formula(object)),
  partial = FALSE, rug = FALSE, rug.sides = "b",
  rug.col = "gray", alpha = 0.2,
  ylim = c("common", "free"), xlabs = NULL,
  ylabs = NULL, main = NULL, term.col = "darkred",
  resid.col = "lightblue", resid.alpha = 0.8,
  resid.size = 1, nrow = NULL, ncol = NULL,
  plots.per.page = 9, one.by.one = FALSE,
  surface.gam = FALSE, polys = NULL,
  polys.scheme = "topo", col.ribbon = "darksalmon",
  col.shaded = "gray", ...)

pe_terms(object,
  what = c("mu", "sigma", "nu", "tau"),
```

```

parameter = NULL, data = NULL, terms = NULL,
envir = environment(formula(object)),
partial = FALSE, rug = FALSE, rug.sides = "b",
rug.col = "gray", alpha = 0.2,
ylim = c("common", "free"), xlabs = NULL,
ylabs = NULL, main = NULL, term.col = "darkred",
resid.col = "lightblue", resid.alpha = 0.8,
resid.size = 1, nrow = NULL, ncol = NULL,
plots.per.page = 9, one.by.one = FALSE,
surface.gam = FALSE, polys = NULL,
polys.scheme = "topo", col.ribbon = "darksalmon",
col.shaded = "gray", ...)

```

Arguments

object	a gamlss fitted model
what	which distribution parameters, i.e. "mu"
parameter	which distribution parameters (equivalent to what)
data	data frame in which variables in object can be found
terms	which terms to plot (default NULL means all terms); a vector passed to <code>lpred(..., type = "terms", terms =*)</code>
envir	environment in which variables in object can be found
partial	logical; should partial residuals be plotted?
rug	add rug plots to the axes
rug.sides	which side the rug "b"=bottom
rug.col	the colour for the rug
alpha	the alpha for the shade
ylim	there are two options here a) "common" and b) "free". The "common" option plots all figures with the same ylim range and therefore allows the viewer to check the relative contribution of each terms compare to the rest. In the 'free' option the limits are computed for each plot separately.
xlabs	the x label
ylabs	the y label
main	title NOT WORKING
term.col	the colour of the line for term
resid.col	the colour of the partial residuals
resid.alpha	The alpha for the partial residuals
resid.size	the size of the partial residuals
nrow	the number of rows in a multiple plot
ncol	the number of rows in a multiple plot
plots.per.page	the number of plots per page in multiple plots

<code>one.by.one</code>	whether to plot the terms one by one
<code>surface.gam</code>	whether to use surface plot if a <code>ga()</code> term is fitted
<code>polys</code>	for GRMF models
<code>polys.scheme</code>	The polygon information file for MRF models
<code>col.ribbon</code>	he colour of the ribbon
<code>col.shaded</code>	he colour of the shaded area
<code>...</code>	for extra argument

Value

A multiple plot

Author(s)

Mikis Stasinopoulos

References

Rigby, R. A. and Stasinopoulos D. M. (2005). Generalized additive models for location, scale and shape,(with discussion), *Appl. Statist.*, **54**, part 3, pp 507-554.

Rigby, R. A., Stasinopoulos, D. M., Heller, G. Z., and De Bastiani, F. (2019) *Distributions for modeling location, scale, and shape: Using GAMLSS in R*, Chapman and Hall/CRC. An older version can be found in <https://www.gamlss.com/>.

Stasinopoulos D. M. Rigby R.A. (2007) Generalized additive models for location scale and shape (GAMLSS) in R. *Journal of Statistical Software*, Vol. **23**, Issue 7, Dec 2007, <https://www.jstatsoft.org/v23/i07/>.

Stasinopoulos D. M., Rigby R.A., Heller G., Voudouris V., and De Bastiani F., (2017) *Flexible Regression and Smoothing: Using GAMLSS in R*, Chapman and Hall/CRC.

Stasinopoulos, M.D., Kneib, T., Klein, N., Mayr, A. and Heller, G.Z., (2024). *Generalized Additive Models for Location, Scale and Shape: A Distributional Regression Approach, with Applications* (Vol. **56**). Cambridge University Press.

(see also <https://www.gamlss.com/>).

See Also

[term.plot](#)

Examples

```
data(aids)
a <- gamlss(y ~ pb(x) + qrt, data = aids, family = NBI)
fitted_terms(a, pages = 1)
```

histSmo_plot	Supporting histSmo()
--------------	----------------------

Description

This function helps to plot density estimates created by the histSmo() function.

Usage

```
histSmo_plot(x, col_fill_bar = gray(0.5), col_bar = "pink",  
             col_line = "darkblue", width_line = 1, title, xlabel)
```

Arguments

x	a histSmo object
col_fill_bar	The fill colour of the bars
col_bar	the colour of the border of the bars
col_line	the colour of the lines
width_line	the width of the lines
title	title if needed
xlabel	x axis label if needed.

Details

This function supports histSmo().

Value

A plot

Author(s)

Mikis Stasinopoulos, Rober Rigby and Fernanda de Bastiani

References

Rigby, R. A., Stasinopoulos, D. M., Heller, G. Z., and De Bastiani, F. (2019) *Distributions for modeling location, scale, and shape: Using GAMLSS in R*, Chapman and Hall/CRC. An older version can be found in <https://www.gamlss.com/>.

Stasinopoulos D. M. Rigby R.A. (2007) Generalized additive models for location scale and shape (GAMLSS) in R. *Journal of Statistical Software*, Vol. **23**, Issue 7, Dec 2007, <https://www.jstatsoft.org/v23/i07/>.

Stasinopoulos D. M., Rigby R.A., Heller G., Voudouris V., and De Bastiani F., (2017) *Flexible Regression and Smoothing: Using GAMLSS in R*, Chapman and Hall/CRC.

Stasinopoulos, M.D., Kneib, T., Klein, N., Mayr, A. and Heller, G.Z., (2024). *Generalized Additive Models for Location, Scale and Shape: A Distributional Regression Approach, with Applications* (Vol. 56). Cambridge University Press.

(see also <https://www.gamlss.com/>).

See Also

[histSmo](#)

Examples

```
a1 <-histSmo(abdom$y)
gg1 <-histSmo_plot(a1)
gg1
```

model_GAIC

Plotting GAIC for GAMLSS models

Description

The function `model_GAIC()` is similar to the function `GAIC.scaled()` of the package **gamlss**. It produces, [for a given set of different fitted models or for a table produced by `chooseDist()`], the scaled Akaike values (see Burnham and Anderson (2002) section 2.9 for a similar concept of the GAIC weights. The plot of the GAIC's should not be interpreted as posterior probabilities of models given the data but can be used for model selection purpose since they produce a scaled ranking of the model using their relative importance i.e. from the worst to the best model.

The function `model_GAIC_lollipop()` is almost identical to `model_GAIC()` but the result is a lollipop plot.

Usage

```
model_GAIC(object, ..., k = 2, c = FALSE, plot = TRUE,
            which = 1, diff.dev = 1000, text.to.show = NULL,
            col = "rosybrown", width = 0.9, horiz = TRUE,
            scale = c("[0,1]", "[max,min]"), title)

model_GAIC_lollipop(object, ..., k = 2, c = FALSE, plot = TRUE,
                    which = 1, diff.dev = 1000, text.to.show = NULL,
                    col = "skyblue", col.point = "blue", pch.point = 19,
                    width = 0.9, horiz = TRUE,
                    scale = c("[0,1]", "[max,min]"), order.val = TRUE, title)

model_TD(..., newdata, plot = TRUE, text.to.show = NULL,
         col = "rosybrown", diff.TD = 1000, width = 0.9, horiz = TRUE,
         scale = c("[0,1]", "[min,max]"), title)

model_TD_lollipop(..., newdata, plot = TRUE, diff.TD = 1000,
```

```
text.to.show = NULL, col = "skyblue",
col.point = "blue", pch.point = 19, width = 0.9,
horiz = TRUE, scale = c("[0,1]", "[min,max]"),
order.val = TRUE, title)
```

Arguments

object	a set of gamlss fitted model(s) or a matrix table produced by chooseDist().
...	gamlss or gamlss2 objects
newdata	the test data
k	the penalty with default k=2
c	whether the corrected AIC, i.e. AICc, should be used, note that it applies only when k=2
plot	whether to plot with default equal TRUE
which	which column of GAIC table to plot
diff.dev, diff.TD	this argument mainly applies only a matrix table produced by chooseDist() and prevents models with a difference in deviance greater than diff.dev from the 'best' model to be considered (or plotted).
text.to.show	if NULL, model_GAIC() shows the model names otherwise the character in this list (the length of which should be equal to the length of models)
col	The colour of the bars (or lines. in the lollipop)
col.point	The colour of the points in the lollipop
pch.point	The points character in the lollipop
width	the width of the bars
horiz	whether to plot the bars horizontally (default) or vertically
scale	the scale of the plot, "[0,1]" plots the AIC's from the worst to the best models in a scale from [0,1]. "[max,min]" plots the AIC's from the worst model to the best model but in the original scale of the AIC's
title	if different title is needed
order.val	whether to order the models from the best to the worst

Details

The option allow the AIC to be plotted from worst to best on a $[0, 1]$ scale using the formula i.e. $(AIC_w - AIC_m)/(AIC_w - AIC_b)$ where the AIC_w and AIC_b are the worst and best AIC, respectively, and AIC_m is the AIC of the current model. If the option scale is set to $[max, min]$ the difference $(AIC_w - AIC_m)$ is plotted.

Value

It returns a bar plot using package **ggplot2**.

Author(s)

Mikis Stasinopoulos, Bob Rigby and Fernanda De Bastiani

References

Burnham K. P. and Anderson D. R (2002). *Model Selection and Multimodel Inference A Practical Information-Theoretic Approach*, Second Edition, Springer-Verlag New York, Inc.

Rigby, R. A. and Stasinopoulos D. M. (2005). Generalized additive models for location, scale and shape,(with discussion), *Appl. Statist.*, **54**, part 3, pp 507-554.

Rigby, R. A., Stasinopoulos, D. M., Heller, G. Z., and De Bastiani, F. (2019) *Distributions for modeling location, scale, and shape: Using GAMLSS in R*, Chapman and Hall/CRC. An older version can be found in <https://www.gamlss.com/>.

Stasinopoulos D. M. Rigby R.A. (2007) Generalized additive models for location scale and shape (GAMLSS) in R. *Journal of Statistical Software*, Vol. **23**, Issue 7, Dec 2007, <https://www.jstatsoft.org/v23/i07/>.

Stasinopoulos D. M., Rigby R.A., Heller G., Voudouris V., and De Bastiani F., (2017) *Flexible Regression and Smoothing: Using GAMLSS in R*, Chapman and Hall/CRC. Stasinopoulos, M.D., Kneib, T., Klein, N., Mayr, A. and Heller, G.Z., (2024). *Generalized Additive Models for Location, Scale and Shape: A Distributional Regression Approach, with Applications* (Vol. **56**). Cambridge University Press.

(see also <https://www.gamlss.com/>).

See Also

[GAIC_scaled](#)

Examples

```
data(abdom)
m1 <- gamlss(y~x, family=NO, data=abdom)
m2 <- gamlss(y~x, sigma.fo=~x, family=NO, data=abdom)
m3 <- gamlss(y~pb(x), sigma.fo=~x, family=NO, data=abdom)
m4 <- gamlss(y~pb(x), sigma.fo=~pb(x), family=NO, data=abdom)

model_GAIC(m1,m2, m3, m4)

MT <- chooseDist(m3)
model_GAIC(MT)
model_GAIC(MT, which=2)
model_GAIC_lollipop(m1,m2, m3, m4)
```

model_pca	<i>Plotting residuals using PCA</i>
-----------	-------------------------------------

Description

The function `model_pca()` plots several GAMLSS residuals using Principal Component Analysis.

Usage

```
model_pca(obj, ..., scale = TRUE, arrow_size = 1.5)
```

Arguments

<code>obj</code>	A gamlss object
<code>...</code>	for extra GAMLSS models
<code>scale</code>	whether to scale the residuals
<code>arrow_size</code>	the arrow sizw in the biplot

Details

The function `model_pca()` plot a `biplot()` plot of the residuals from different models. It uses Principal Component Analysis in the residuals of different models and plots the results.

Value

A biplot of the first two components is plotted.

Author(s)

Mikis Stasinopoulos, Bob Rigby and Fernanda De Bastiani

References

Rigby, R. A., Stasinopoulos, D. M., Heller, G. Z., and De Bastiani, F. (2019) *Distributions for modeling location, scale, and shape: Using GAMLSS in R*, Chapman and Hall/CRC. An older version can be found in <https://www.gamlss.com/>.

Stasinopoulos D. M. Rigby R.A. (2007) Generalized additive models for location scale and shape (GAMLSS) in R. *Journal of Statistical Software*, Vol. **23**, Issue 7, Dec 2007, <https://www.jstatsoft.org/v23/i07/>.

Stasinopoulos D. M., Rigby R.A., Heller G., Voudouris V., and De Bastiani F., (2017) *Flexible Regression and Smoothing: Using GAMLSS in R*, Chapman and Hall/CRC.

Stasinopoulos, M.D., Kneib, T., Klein, N., Mayr, A. and Heller, G.Z., (2024). *Generalized Additive Models for Location, Scale and Shape: A Distributional Regression Approach, with Applications* (Vol. **56**). Cambridge University Press.

(see also <https://www.gamlss.com/>).

See Also

[gamlss](#), [resid_index](#)

Examples

```
m1 <- gamlss(y~x, data=abdom)
m2 <- gamlss(y~pb(x), data=abdom)
m3 <- gamlss(y~pb(x), sigma.fo=~pb(x), data=abdom)
model_pca(m1,m2,m3)
```

moment_bucket

Moment bucket plot

Description

A bucket plot is a graphical way to check the skewness and kurtosis of a continuous variable or the residuals of a fitted GAMLSS model. It plots the transformed moment skewness and transformed moment kurtosis of the variable (or residuals) together with a cloud of points obtained using a non-parametric bootstrap from the original variable (or residuals). It also provides a graphical way of performing the Jarque-Bera test (Jarque and Bera, 1980).

There are two functions here:

- i) `moment_bucket()` for a single bucket plot. Note that `model_mom_bucket()` and `moment_bucket()` are synonymous.
- ii) `moment_bucket_wrap()` for multiple bucket plots cut according to terms in the model.

Usage

```
moment_bucket(x, ..., weights = NULL, no_bootstrap = 99,
              col_bootstrap = hcl.colors(length.obj, palette = "Set 2"),
              alpha_bootstrap = 1, text_to_show = NULL,
              cex_text = 5, col_text = "black", colour_bucket = FALSE,
              line_width = 0.5, col_JB_test = gray(.7), alpha_JB_test = .1)

model_mom_bucket(x, ..., weights = NULL, no_bootstrap = 99,
                 col_bootstrap = hcl.colors(length.obj, palette = "Set 2"),
                 alpha_bootstrap = 1, text_to_show = NULL,
                 cex_text = 5, col_text = "black", colour_bucket = FALSE,
                 line_width = 0.5, col_JB_test = gray(.7), alpha_JB_test = .1)

moment_bucket_wrap(x, ..., weights = NULL, xvar = NULL, n_inter = 4,
                  no_bootstrap = 99,
                  col_bootstrap = hcl.colors(length.obj, palette = "Set 2"),
                  alpha_bootstrap = 1, text_to_show = NULL,
                  check_overlap_text = FALSE, cex_text = 5,
                  col_text = "black", colour_bucket = FALSE,
                  col_JB_test = gray(.7), alpha_JB_test = .1)
```

Arguments

x	x should be a continuous vector of a GAMLSS fitted model.
...	this for more that one continuous vectors or fitted models
weights	if priors weights are needed
no_bootstrap	the number of bootstrap samples for the cloud around the point of skewness and kurtosis.
col_bootstrap	The colour of the bootstrap samples
alpha_bootstrap	The transparency parameter of the bootstrap samples.
text_to_show	what text to show in the plots, default the names of vectors ot models
cex_text	the character size of the text
col_text	the colour of the text
colour_bucket	whether colour or gray lines in the bucket
line_width	the line width
xvar	the x term
n_inter	how many intervals needed
check_overlap_text	whether to check overlapping text
col_JB_test	the colour for the Jarque-Bera test
alpha_JB_test	the transparency constant for the Jarque-Bera test

Value

A plot displaying the transformed moment skewness and transformed moment kurtosis of the sample or residual of a model.

Note

The bucket plot provides an additional residual diagnostic tool that can be used for fitted model checking, alongside other diagnostic tools, for example worm plots, and Q (and Z) statistics.

Author(s)

Mikis Stasinopoulos, Bob Rigby and Fernanda De Bastiani

References

- De Bastiani, F., Stasinopoulos, D. M., Rigby, R. A., Heller, G. Z., and Lucas A. (2022) Bucket Plot: A Visual Tool for Skewness and Kurtosis Comparisons. To be published.
- Rigby, R. A. and Stasinopoulos D. M. (2005). Generalized additive models for location, scale and shape,(with discussion), *Appl. Statist.*, **54**, part 3, pp 507-554.
- Rigby, R. A., Stasinopoulos, D. M., Heller, G. Z., and De Bastiani, F. (2019) *Distributions for modeling location, scale, and shape: Using GAMLSS in R*, Chapman and Hall/CRC. doi:10.1201/9780429298547 An older version can be found in <https://www.gamlss.com/>.

Stasinopoulos D. M., Rigby R.A. (2007) Generalized additive models for location scale and shape (GAMLSS) in R. *Journal of Statistical Software*, Vol. **23**, Issue 7, Dec 2007, doi:[10.18637/jss.v023.i07](https://doi.org/10.18637/jss.v023.i07).

Stasinopoulos D. M., Rigby R.A., Heller G., Voudouris V., and De Bastiani F., (2017) *Flexible Regression and Smoothing: Using GAMLSS in R*, Chapman and Hall/CRC. doi:[10.1201/b21973](https://doi.org/10.1201/b21973)

Stasinopoulos, M. D., Rigby, R. A., and De Bastiani F., (2018) GAMLSS: a distributional regression approach, *Statistical Modelling*, Vol. **18**, pp, 248-273, SAGE Publications Sage India: New Delhi, India.

Stasinopoulos, M.D., Kneib, T., Klein, N., Mayr, A. and Heller, G.Z., (2024). *Generalized Additive Models for Location, Scale and Shape: A Distributional Regression Approach, with Applications* (Vol. **56**). Cambridge University Press.

(see also <https://www.gamlss.com/>).

See Also

[wp](#), [Q.stats](#)

Examples

```
m1 <- gamlss(R~pb(Fl)+pb(A), data=rent, family=GA)
moment_bucket(m1)
moment_bucket_wrap(m1, xvar=rent$A)
```

moment_gray_half

Functions to create the background for the bucket plots

Description

The functions plot the moment transformed skewness and moment transformed kurtosis of five important 4-parameter distributions in GAMLSS.

Usage

```
moment_gray_half(legend = FALSE)

moment_gray_both(line_width = 1)

moment_colour_half(legend = TRUE)

moment_colour_both(legend = TRUE, line_width = 1)

centile_colour_half(type = c("tail", "central"), legend = TRUE,
                    line_width = 1)

centile_colour_both(type = c("tail", "central"), legend = TRUE,
                   line_width = 1)

centile_gray_both(type = c("tail", "central"), legend = TRUE,
                 line_width = 0.5)
```

Arguments

legend	whether legend is required
line_width	line width
type	whether to plot ‘tail’ or ‘central’ skewness and kurtosis.

Details

The functions are described in Rigby *et al* (2019)

Value

A plot is created.

Note

The functions are use by the bucket plot function `model_mom_bucket()` to create the background of the bucket plots.

Author(s)

Mikis Stasinopoulos, Bob Rigby and Fernanda de Bastiani

References

- Rigby, R. A. and Stasinopoulos D. M. (2005). Generalized additive models for location, scale and shape,(with discussion), *Appl. Statist.*, **54**, part 3, pp 507-554.
- Rigby, R. A., Stasinopoulos, D. M., Heller, G. Z., and De Bastiani, F. (2019) *Distributions for modeling location, scale, and shape: Using GAMLSS in R*, Chapman and Hall/CRC. doi:10.1201/9780429298547 An older version can be found in <https://www.gamlss.com/>.
- Stasinopoulos D. M. Rigby R.A. (2007) Generalized additive models for location scale and shape (GAMLSS) in R. *Journal of Statistical Software*, Vol. **23**, Issue 7, Dec 2007, doi:10.18637/jss.v023.i07.
- Stasinopoulos D. M., Rigby R.A., Heller G., Voudouris V., and De Bastiani F., (2017) *Flexible Regression and Smoothing: Using GAMLSS in R*, Chapman and Hall/CRC. doi:10.1201/b21973
- Stasinopoulos, M.D., Kneib, T., Klein, N., Mayr, A. and Heller, G.Z., (2024). *Generalized Additive Models for Location, Scale and Shape: A Distributional Regression Approach, with Applications* (Vol. **56**). Cambridge University Press.
- (see also <https://www.gamlss.com/>).

See Also

[momentSK](#), [centileSK](#)

Examples

```
moment_gray_half()
moment_gray_both()
moment_colour_half()
moment_colour_both()

centile_colour_both()
centile_gray_both()
centile_colour_half()
```

pcr_coef_path

Plotting the fitted path of a PCR model.

Description

This function is similar to the function `plot.PCR()` which is used to plot the path of a fitted principal componet regression model, fitted using the function `fitPCR()` of the package **gamlss.foreach**.

Usage

```
pcr_coef_path(x, legend=FALSE, plot=TRUE)

pcr_path(x, parameter = c("mu", "sigma", "nu", "tau"),
         legend = FALSE, plot = TRUE)
```

Arguments

<code>x</code>	a fitted PCR object (or a fitted GAMLSS object for function <code>pcr_path()</code>)
<code>legend</code>	whether legent is needed
<code>plot</code>	whether to plot the path
<code>parameter</code>	which GAMLSS parameter, between "mu", "sigma", "nu", "tau"

Value

A gg-plot.

Author(s)

Mikis Stasinopoulos, Bob Rigby and Fernanda De Bastiani

References

Rigby, R. A. and Stasinopoulos D. M. (2005). Generalized additive models for location, scale and shape,(with discussion), *Appl. Statist.*, **54**, part 3, pp 507-554.

Rigby, R. A., Stasinopoulos, D. M., Heller, G. Z., and De Bastiani, F. (2019) *Distributions for modeling location, scale, and shape: Using GAMLSS in R*, Chapman and Hall/CRC. An older version can be found in <https://www.gamlss.com/>.

Stasinopoulos D. M. Rigby R.A. (2007) Generalized additive models for location scale and shape (GAMLSS) in R. *Journal of Statistical Software*, Vol. **23**, Issue 7, Dec 2007, <https://www.jstatsoft.org/v23/i07/>.

Stasinopoulos D. M., Rigby R.A., Heller G., Voudouris V., and De Bastiani F., (2017) *Flexible Regression and Smoothing: Using GAMLSS in R*, Chapman and Hall/CRC.

Stasinopoulos, M.D., Kneib, T., Klein, N., Mayr, A. and Heller, G.Z., (2024). *Generalized Additive Models for Location, Scale and Shape: A Distributional Regression Approach, with Applications* (Vol. **56**). Cambridge University Press.

(see also <https://www.gamlss.com/>).

See Also

[plot.PCR](#)

Examples

```
library(gamlss.foreach)
library(glmnet)
library(ggplot2)
data(QuickStartExample)
attach(QuickStartExample)
hist(y, main="(a)")
if (is.null(rownames(x)))
  colnames(x) <- paste0("X", seq(1:dim(x)[2]))
# fitting
MM<- fitPCR(x,y, k=log(100))
pp<-pcr_coef_path(MM)
pp+ ggplot2::geom_vline(xintercept = MM$pc, colour = "gray")
# using gamlss
m1 <- gamlss(y~pcr(x=x))
pcr_path(m1)
```

Description

There are several functions here for plotting partial effect of terms in a distributional parameter GAMLSS model. Those functions help the interpretation of a GAMLSS model.

In the hard of it there are the two functions;

i) `pe_1_paramater()` and

ii) `pe_2_paramater()`.

The first plots the effect of one term in the data keeping the rest of the term at specified values (scenarios). The second plots the effect of two term keeping the rest of the term at specified values. The default scenarios are specified using the argument `how.scenario`.

The function `pe_param()` calls either `pe_1_paramater()` or `pe_2_paramater()` depending on the argument `term`. That is whether the argument is a single argument, e.g. "age", or a double one e.g. `c("age", "gender")`.

Similarly with the function `pe_eta()` which is almost identical to `pe_param()` but instead of plotting terms for the parameters (i.e. μ or σ) it plots terms for the predictors (i.e. η_μ or η_σ).

The functions

i) `pa_param_grid()` and ii) `pa_eta_grid()`

produce multiple plots on a grid.

Usage

```
pe_1_parameter(obj = NULL, term = NULL, data = NULL,
               n.points = 100, parameter = c("mu", "sigma", "nu", "tau"),
               type = c("parameter", "eta"), scenario = list(),
               how.scenario = c("median", "last", "fixed"),
               centered.from = c("mean", "median", "none", "rlevel"),
               col = "darkblue", linewidth = 1.3, name.obj = NULL,
               data.plot = FALSE, data.col = "lightblue",
               data.size = 0.1, data.alpha = 0.9,
               rug.plot = TRUE, rug.col = "gray", rug.size = 0.5,
               factor.size = 15, ylim = NULL, plot = TRUE, title)
```

```
pe_2_parameter(obj = NULL, terms = NULL, data = NULL,
               n.points = 100, parameter = c("mu", "sigma", "nu", "tau"),
               type = c("parameter", "eta"),
               how.scenario = c("median", "last", "fixed"),
               scenario = list(),
               centered.from = c("mean", "median", "none"),
               col = "darkblue", linewidth = 1.3, data.plot = FALSE,
               data.col = "lightblue", data.size = 0.1,
               data.alpha = 0.9, bins = 30,
               filled = FALSE, name.obj = NULL, plot = TRUE, title)
```

```
pe_param(obj = NULL, term = NULL, data = NULL, n.points = 100,
         parameter = c("mu", "sigma", "nu", "tau"),
         type = c("parameter", "eta"),
```

```

scenario = list(), how.scenario = c("median", "last", "fixed"),
col = "darkblue", linewidth = 1.3, name.obj = NULL,
rug.plot = TRUE, rug.col = "gray", rug.size = 0.5,
data.plot = FALSE, data.col = "lightblue", data.size = 0.1,
factor.size = 15, data.alpha = 0.9, bins = 30,
filled = FALSE, ylim = NULL, title, ...)

pe_eta(obj = NULL, term = NULL, data = NULL, n.points = 100,
parameter = c("mu", "sigma", "nu", "tau"),
scenario = list(), how.scenario = c("median", "last", "fixed"),
col = "darkblue", linewidth = 1.3, name.obj = NULL,
rug.plot = TRUE, rug.col = "gray", rug.size = 0.5,
data.plot = FALSE, data.col = "lightblue", data.size = 0.1,
factor.size = 15, data.alpha = 0.9, bins = 30,
filled = FALSE, ylim = NULL, title, ...)

pe_param_grid(model, terms, maxcol = 2, maxrow = 3, ylim = NULL, ...)

pe_eta_grid(model, terms, maxcol = 2, maxrow = 3, ylim = NULL, ...)

```

Arguments

obj	a GAMLSS fitted object
model	a GAMLSS fitted model
term	the model term we want to investigate can be one i.e. "Fl" or two c("Fl", "A")
terms	This argument is for functions <code>pa_param_grid()</code> and <code>pa_eta_grid()</code> . It should be a list of model terms, for example, <code>list(c("Fl", "A"), "H", "loc")</code>
data	the data frame used otherwise it takes it from the fitted model
n.points	the number of points for the evaluation of the term
parameter	the distribution parameter in which the term is fitted
type	this is for functions <code>pe_1_parameter()</code> and <code>pe_2_parameter()</code> , it allows to specify the "parameter" or its prediction, "eta"
how.scenario	how to set all the rest of the continuous terms in the model if are not specified by scenario. The alternatives are for "median", "last" (for time series) or "fixed"
scenario	this can be a list of values for the rest of the terms in the model for the specific distribution parameter
plot	whether to plot the result or just get the created data.frame
col	the colour of the partial effect of the term
linewidth	the size of the line of partial effect of the term
bins	the number of binds for the contour plot
filled	whether to display the values in the contour
title	the title if different from the default

name.obj	this is a way to pass the name of the object
maxcol	the maximum columns in the grid plot
maxrow	the maximum rowss in the grid plot
data.plot	whether to plot the data
rug.plot	whether to print the rug bellow the figure
rug.size	the size of the rug
rug.col	the colour of the rug
data.col	the color of the data points
data.size	the size of the data points
data.alpha	the trnsparance constant of the data points
factor.size	the size of the symbol if a factor is plotted
ylim	if a common y limit is required
centered.from	whether to centered the result
...	for passing argument from the function pe_param_grid to the function pe_param

Details

The functions `pe_param()` and `pe_param_grid()` can be used to help the use the interpretation of a GAMLSS model. The functions `pe_param()` provides the partial effect of one or two terms of a specified parameter of the distribution while the rest of the terms in the model are set on specific values or scenarios. The function `pe_param()` calls `pe_1param()` if the argument terms is one i.e. "Fl" or the function `pe_2param()` if the terms are two i.e. `c("Fl","A")`. The `pe_param_grid()` plots multiple plots specified by the list used in the term argument.

Similar functions are `pe_quantile()` and `pe_moment()`.

Value

It is plotting the partial effect or is producing the resulting function

Note

The data if is not declared is taken for from the fitted model. So the `data.frame` used for fitting the model should exist in your working space. Variables in the data set not used in the fit could slow down the functions because they go thought a loop containg all variables (columns) in the data.

A term do not have to be in the model, (in which case his contribution will be flat), but it has to be in the data.

Author(s)

Mikis Stasinopulos

References

- Rigby, R. A. and Stasinopoulos D. M. (2005). Generalized additive models for location, scale and shape,(with discussion), *Appl. Statist.*, **54**, part 3, pp 507-554.
- Rigby, R. A., Stasinopoulos, D. M., Heller, G. Z., and De Bastiani, F. (2019) *Distributions for modeling location, scale, and shape: Using GAMLSS in R*, Chapman and Hall/CRC. An older version can be found in <https://www.gamlss.com/>.
- Stasinopoulos D. M. Rigby R.A. (2007) Generalized additive models for location scale and shape (GAMLSS) in R. *Journal of Statistical Software*, Vol. **23**, Issue 7, Dec 2007, <https://www.jstatsoft.org/v23/i07/>.
- Stasinopoulos D. M., Rigby R.A., Heller G., Voudouris V., and De Bastiani F., (2017) *Flexible Regression and Smoothing: Using GAMLSS in R*, Chapman and Hall/CRC.
- Stasinopoulos, M.D., Kneib, T., Klein, N., Mayr, A. and Heller, G.Z., (2024). *Generalized Additive Models for Location, Scale and Shape: A Distributional Regression Approach, with Applications* (Vol. **56**). Cambridge University Press.
- (see also <https://www.gamlss.com/>).

See Also

[getPEF](#)

Examples

```
library(gamlss)
m1 <- gamlss(R~pb(F1)+pb(A)+loc+H, data=rent, family=GA)
pe_param(m1, "A")
pe_param(m1, c("F1","A"), filled=TRUE)
#pe_param_grid(m1, list(c("F1","A"), c("H","loc")), filled=TRUE)
# the terms are additive no interaction
```

pe_pdf

Partial Effect of a term on the response distribution

Description

The function `pe_pdf()` plots the partial effect that a specified term has on the distribution of the response.

The function `pe_pdf_grid()` plot multiple plots on the same page.

Usage

```
pe_pdf(obj = NULL, term = NULL, from = NULL, to = NULL,
       y.grid.points = 100, x.grid.points = 10, x.values,
       data = NULL, scale = NULL, how = c("median", "last"),
       scenario = list(), linewidth = 0.1, horizontal = TRUE,
       col.fill = hcl.colors(lqq, palette = "viridis"),
```

```

        alpha = 0.6, xlim = NULL, title)

pe_pdf_grid(model, terms, maxcol = 2, maxrow = 3, ...)

```

Arguments

obj, model	A GAMLSS object
term	The model term
terms	The model terms, more than one for pe_pdf_grid().
from	start from
to	end to
y.grid.points	in how many points the pdf should be evaluates
x.grid.points	in how many points the terms should be plotted
x.values	possible x values
data	The data used for modelling
scale	This is a very important value for plotting correctly the fitted distributions. If the default values it is not working please try different values
how	How to fix the rest of the variables. For continuous it takes the median for factor the level with the highest frequency.
scenario	Alternatively scenario for fixing the values.
linewidth	the size of the pdf line
horizontal	whether to plot the partial pdf on the x-axis and the x on the y-axis or opposite
col.fill	how to fill the pdf body
alpha	the transparency factor
xlim	the limits for plotting x-axis
title	whether to use a different title from the default one
maxcol	maximum of columns in the grid for pe_pdf_grid()
maxrow	maximum of rows on the grid for pe_pdf_grid()
...	extra argument to be passed from pe_pdf() to pe_pdf_grid()

Details

The function `pe_pdf()` is one of the functions designed to help the user to interpret the GAMLSS model. Provides the partial effect that one of the continuous terms has on the distribution of the response while the rest of the variables in the model are set on specific values or scenarios. Other similar functions are `pe_param()`, `pe_moment()` and `pe_quantile()`.

Value

A plot of the conditional distribution given the term

Author(s)

Mikis Stasinopoulos, Rober Rigby and Fernanda de Bastiani

References

Rigby, R. A., Stasinopoulos, D. M., Heller, G. Z., and De Bastiani, F. (2019) *Distributions for modeling location, scale, and shape: Using GAMLSS in R*, Chapman and Hall/CRC. An older version can be found in <https://www.gamlss.com/>.

Stasinopoulos D. M. Rigby R.A. (2007) Generalized additive models for location scale and shape (GAMLSS) in R. *Journal of Statistical Software*, Vol. **23**, Issue 7, Dec 2007, <https://www.jstatsoft.org/v23/i07/>.

Stasinopoulos D. M., Rigby R.A., Heller G., Voudouris V., and De Bastiani F., (2017) *Flexible Regression and Smoothing: Using GAMLSS in R*, Chapman and Hall/CRC.

Stasinopoulos, M.D., Kneib, T., Klein, N., Mayr, A. and Heller, G.Z., (2024). *Generalized Additive Models for Location, Scale and Shape: A Distributional Regression Approach, with Applications* (Vol. **56**). Cambridge University Press.

(see also <https://www.gamlss.com/>).

See Also

[pe_param](#)

Examples

```
m1 <- gamlss(R~pb(F1)+pb(A)+loc+H, data=rent, family=GA)
pe_pdf(m1, "A")
pe_pdf(m1, "A")
pe_pdf(m1, "A", horizontal=FALSE)
pe_pdf_grid(m1, c("F1", "A", "H", "loc"))
```

prof_term

Plotting the profile deviance of one fitted term

Description

Plotting the profile deviance of a fitted term in GAMLSS

Usage

```
prof_term(model = NULL, criterion = c("GD", "GAIC"),
  penalty = 2.5, from = NULL, to = NULL,
  step = NULL, length = 7, xlabel = NULL, plot = TRUE,
  perc = 95, start.prev = TRUE, line.col = "darkgreen",
  dash.line.type = 3, dash.line.size = 0.8, text.size = 5, title)
```

Arguments

model	a GAMLSS fitted model
criterion	whether Global deviance or GAIC
penalty	the penalty k for GAIC
from	start from
to	finish at
step	using step
length	if the step is left NULL then length is considered for evaluating the grid for the parameter. It has a default value of 11.
xlabel	if a x label is required
plot	whether to plot the function
perc	what percentage confidence interval is required
start.prev	whether to start from the previous fitted model parameters values or not (default is TRUE)
line.col	the colour of the plotting line
dash.line.type	the type of vertical dash line for CI's
dash.line.size	The size of the dash lines
text.size	the size of text
title	the title

Details

This function is the ggplot2 version of the original `prof.term()` function.

Value

creates a plot

Author(s)

Mikis Stasinopoulos

References

- Rigby, R. A. and Stasinopoulos D. M. (2005). Generalized additive models for location, scale and shape,(with discussion), *Appl. Statist.*, **54**, part 3, pp 507-554.
- Rigby, R. A., Stasinopoulos, D. M., Heller, G. Z., and De Bastiani, F. (2019) *Distributions for modeling location, scale, and shape: Using GAMLSS in R*, Chapman and Hall/CRC. An older version can be found in <https://www.gamlss.com/>.
- Stasinopoulos D. M. Rigby R.A. (2007) Generalized additive models for location scale and shape (GAMLSS) in R. *Journal of Statistical Software*, Vol. **23**, Issue 7, Dec 2007, <https://www.jstatsoft.org/v23/i07/>.
- Stasinopoulos D. M., Rigby R.A., Heller G., Voudouris V., and De Bastiani F., (2017) *Flexible Regression and Smoothing: Using GAMLSS in R*, Chapman and Hall/CRC.

Stasinopoulos, M.D., Kneib, T., Klein, N., Mayr, A. and Heller, G.Z., (2024). *Generalized Additive Models for Location, Scale and Shape: A Distributional Regression Approach, with Applications* (Vol. 56). Cambridge University Press.

(see also <https://www.gamlss.com/>).

See Also

[prof.term](#)

Examples

```
data(aids)
# fitting a linear model
gamlss(y~x+qrt,family=NBI,data=aids)
# testing the linear beta parameter
mod<-quote(gamlss(y ~ offset(this * x) + qrt, data = aids, family = NBI))
prof_term(mod, from = .06, to=0.13)
```

resid_density

Density of the residuals in a GAMLLSS model

Description

The function `resid_density()` plots an histogram and a density estimator of the normalised quantile residuals from a fitted GAMLLSS model. The function `model_density()` plots density estimators of the normalised quantile residuals from more than one fitted GAMLLSS models.

Usage

```
resid_density(obj, resid, hist.col = "black", hist.fill = "white",
              dens.fill = "#FF6666", title)
model_density(obj, ..., title)
```

Arguments

<code>obj</code>	The function needs a GAMLLSS fitted model or
<code>resid</code>	any standardised residual
<code>hist.col</code>	The colour of the border of the histogram
<code>hist.fill</code>	The colour of the histogram
<code>dens.fill</code>	the colour of the density
<code>title</code>	A title if needed
<code>...</code>	extra GAMLLSS models

Details

This function `resid_density()` is a density plot (similar to of the four plots produced when the plotting function `plot.gamlss()` is used within the **gamlss** package. It uses plotting function from the **ggplot2** package.

Value

A density plot of the residuals is produced.

Author(s)

Mikis Stasinopoulos, Bob Rigby and Fernanda De Bastiani

References

Rigby, R. A., Stasinopoulos, D. M., Heller, G. Z., and De Bastiani, F. (2019) *Distributions for modeling location, scale, and shape: Using GAMLSS in R*, Chapman and Hall/CRC. An older version can be found in <https://www.gamlss.com/>.

Stasinopoulos D. M. Rigby R.A. (2007) Generalized additive models for location scale and shape (GAMLSS) in R. *Journal of Statistical Software*, Vol. **23**, Issue 7, Dec 2007, <https://www.jstatsoft.org/v23/i07/>.

Stasinopoulos D. M., Rigby R.A., Heller G., Voudouris V., and De Bastiani F., (2017) *Flexible Regression and Smoothing: Using GAMLSS in R*, Chapman and Hall/CRC.

Stasinopoulos, M.D., Kneib, T., Klein, N., Mayr, A. and Heller, G.Z., (2024). *Generalized Additive Models for Location, Scale and Shape: A Distributional Regression Approach, with Applications* (Vol. **56**). Cambridge University Press.

(see also <https://www.gamlss.com/>).

See Also

[plot.gamlss](#)

Examples

```
data(abdom)
a<-gamlss(y~pb(x),family=L0,data=abdom)
b<-gamlss(y~pb(x),family=N0,data=abdom)
resid_density(a)
model_density(a,b)
```

resid_dtop

Detrended Transformed Owen's Plot and ECDF for the residuals

Description

The function `resid_dtop()` provides single de-trended transformed Owen's plot, Owen (1995), for a GAMLSS fitted objects or any other residual vector (suitable standardised). This is a diagnostic tool for checking whether the normalised quantile residuals are coming from a normal distribution or not. This could be true if the horizontal line is within the confidence intervals.

The function `resid_ecdf()` provides the empirical cumulative distribution function of the residuals.

The function `y_ecdf()` provides the empirical cumulative distribution function of any numerical vector `y`.

The function `y_epdf()` provides the empirical probability distribution function of any numerical vector `y`. The plot could be uninterpretable if there are no repetitions in the number of distinct values for `y`.

The function `y_epit()` calculates the empirical probability integral transform, PIT, for a continuous vector in the data.

Usage

```
resid_dtop(obj, resid, type = c("Owen", "JW"), conf.level = c("95", "99"),
           value = 2, points.col = "steelblue4",
           check_overlap = TRUE, title, ylim, ...)

resid_ecdf(obj, resid, type = c("Owen", "JW"), conf.level = c("95", "99"),
           value = 2, points.col = "steelblue4",
           check_overlap = TRUE, show.outliers = TRUE, title, ...)

y_ecdf(y, type = c("Owen", "JW"), conf.level = c("95", "99"), value = 2,
       points.col = "steelblue4", check_overlap = TRUE,
       show.outliers = FALSE, from, to, title, ...)

y_epdf(y, plot = TRUE, colour=gray(.5))

y_epit(y)
```

Arguments

<code>obj</code>	A GAMLSS fitted model
<code>resid</code>	if the object is not specified the residual vector can be given here
<code>y</code>	a numeric vector
<code>type</code>	whether to use Owen (1995) or Jager and Wellner (2004) approximate formula
<code>conf.level</code>	95% (default) or 99% percent confidence interval for the plots
<code>value</code>	cut of point for large residuals
<code>points.col</code>	the colour of the points in the plot
<code>check_overlap</code>	to check for overlap when plotting the observation numbers
<code>title</code>	required title
<code>show.outliers</code>	whether to show the number of an outlier observation
<code>ylim</code>	if the y limit should be different from the default <code>max(y)+.1</code>
<code>from</code>	where to start the ecdf
<code>to</code>	where to finish the ecdf
<code>plot</code>	whether to plot or save the data as a <code>data.frame</code>
<code>colour</code>	the colour of the spikes in the <code>y_epdf</code>
<code>...</code>	further arguments

Value

A ggplot is generated

Author(s)

Mikis Stasinopoulos, Bob Rigby and Fernanda de Bastiani

References

Jager, L. and Wellner, J. A (2004) A new goodness of fit test: the reversed Berk-Jones statistic, University of Washington, Department of Statistics, Technical report 443.

Owen A. B. (1995) Nonparametric Confidence Bands for a Distribution Function. *Journal of the American Statistical Association* Vol. 90, No 430, pp. 516-521.

Rigby, R. A. and Stasinopoulos D. M. (2005). Generalized additive models for location, scale and shape,(with discussion), *Appl. Statist.*, 54, part 3, 1-38.

Rigby, R. A., Stasinopoulos, D. M., Heller, G. Z., and De Bastiani, F. (2019) *Distributions for modeling location, scale, and shape: Using GAMLSS in R*, Chapman and Hall/CRC. An older version can be found in <https://www.gamlss.com/>.

Stasinopoulos D. M. Rigby R.A. (2007) Generalized additive models for location scale and shape (GAMLSS) in R. *Journal of Statistical Software*, Vol. 23, Issue 7, Dec 2007, <https://www.jstatsoft.org/v23/i07/>.

Stasinopoulos D. M., Rigby R.A., Heller G., Voudouris V., and De Bastiani F., (2017) *Flexible Regression and Smoothing: Using GAMLSS in R*, Chapman and Hall/CRC.

Stasinopoulos, M.D., Kneib, T., Klein, N., Mayr, A. and Heller, G.Z., (2024). *Generalized Additive Models for Location, Scale and Shape: A Distributional Regression Approach, with Applications* (Vol. 56). Cambridge University Press.

(see also <https://www.gamlss.com/>).

See Also

[resid_wp](#)

Examples

```
library(ggplot2)
data(abdom)
a<-gamlss(y~pb(x),sigma.fo=~pb(x,1),family=L0,data=abdom)
resid_dtop(a)
resid_ecdf(a)+ stat_function(fun = pN0, args=list(mu=0, sigma=1))
# create a gamma distributed random variable
y <- rGA(1000, mu=3, sigma=1)
gp<- y_ecdf(y)
gp + stat_function(fun = pGA, args=list(mu=3, sigma=1))
```

resid_index

*Residual plots***Description**

The function `resid_index()` is plotting the residuals of a GAMLSS fitted model (or any other suitable standardised residual) against the observation number index.

The function `resid_mu()` plots the residuals against fitted values for μ .

The function `resid_median()` plots the residuals against fitted median values.

The function `resid_param()` plots the residuals against any of the GAMLSS fitted parameters, e.g. μ , σ , ν , or τ .

The function `resid_quantile()` plots the residuals against any fitted quantile.

The function `resid_xvar()` plots the residuals against any explanatory term.

The function `resid_plots()` produces a plot similar to the one that the function `plot()` produce for a GAMLSS model in package **gamlss**. This is, four plots: a) `resid_index()` (b) `resid_mu()`, (c) `resid_density()` and (d) `resid_qqplot()`.

Residuals above (or below) certain specified value are identified.

Usage

```
resid_index(obj, resid, plot = TRUE, value = 2, title, annotate = TRUE,
            no.lines = FALSE)
```

```
resid_mu(obj, resid, plot = TRUE, value = 2, title, annotate = TRUE)
```

```
resid_median(obj, resid, plot = TRUE, value = 3, title,
             annotate = TRUE)
```

```
resid_param(obj, param = c("mu", "sigma", "nu", "tau"), title,
            line.col = "darkred", point.col = "steelblue4",
            point.shape = 20)
```

```
resid_quantile(obj, quantile = 0.5, title, newdata,
               line.col = "darkred", point.col = "steelblue4",
               point.shape = 20)
```

```
resid_plots(obj, theme = c("original", "ts", "new", "ecdf"), value = 3)
```

```
resid_xvar(obj, xvar, plot = TRUE, value = 2, title, annotate = TRUE)
```

Arguments

<code>obj</code>	a GAMLSS object
<code>resid</code>	or any other suitable standardised residual vector.

xvar	a continuous explanatory variable
plot	whether to plot the result
param	which GAMLSS parameter mu, sigma, nu, or tau
value	the cut off value for the identification of very large or very small residuals
annotate	whether the threshold annotation should appear or not
line.col	the colour of the line
point.col	the colour of the points
point.shape	the shape of the points
title	a title of the plot if needed
theme	what type of plots should resid_plots() used: "original" is like using plot.gamlss(), "ts" is like using plot.gamlss(ts="TRUE") (not implemented yet), "new" it uses (a) resid_index(), (b) resid_density(), (c) resid_wp() and (d) resid_dtop().
no.lines	this option allows to hide the horizontal lines so the resulting gg-plot can be used later with say facet_wrap() see example
newdata	whether the evaluation should be in newdata or the old data points
quantile	which quantile? default the median (0.50).

Value

A plot of the residuals is returned.

Author(s)

Mikis Stasinopoulos, Bob Rigby and Fernanda De Bastiani

References

- Rigby, R. A. and Stasinopoulos D. M. (2005). Generalized additive models for location, scale and shape,(with discussion), *Appl. Statist.*, **54**, part 3, pp 507-554.
- Rigby, R. A., Stasinopoulos, D. M., Heller, G. Z., and De Bastiani, F. (2019) *Distributions for modeling location, scale, and shape: Using GAMLSS in R*, Chapman and Hall/CRC, doi:10.1201/9780429298547. An older version can be found in <https://www.gamlss.com/>.
- Stasinopoulos D. M. Rigby R.A. (2007) Generalized additive models for location scale and shape (GAMLSS) in R. *Journal of Statistical Software*, Vol. **23**, Issue 7, Dec 2007, doi:10.18637/jss.v023.i07.
- Stasinopoulos D. M., Rigby R.A., Heller G., Voudouris V., and De Bastiani F., (2017) *Flexible Regression and Smoothing: Using GAMLSS in R*, Chapman and Hall/CRC. doi:10.1201/b21973
- Stasinopoulos, M. D., Rigby, R. A., and De Bastiani F., (2018) GAMLSS: a distributional regression approach, *Statistical Modelling*, Vol. **18**, pp, 248-273, SAGE Publications Sage India: New Delhi, India.
- Stasinopoulos, M.D., Kneib, T., Klein, N., Mayr, A. and Heller, G.Z., (2024). *Generalized Additive Models for Location, Scale and Shape: A Distributional Regression Approach, with Applications* (Vol. **56**). Cambridge University Press.
- (see also <https://www.gamlss.com/>).

See Also

[gamlss](#), [plot.gamlss](#)

Examples

```
library(ggplot2)
data(rent)
r1<-gamlss(R~pb(F1)+pb(A)+H+loc,family=GA,data=rent)
resid_index(r1)
resid_mu(r1)
resid_median(r1)
resid_param(r1)
resid_quantile(r1)
resid_xvar(r1, A)
resid_plots(r1)
resid_index(r1, no.lines=TRUE)+facet_wrap(~ cut_number(rent$A, 6))
```

resid_qqplot

QQ-plot of the residuals of a GAMLSS model

Description

The function `resid_qqplot()` produces a single QQ-plot of the residuals from a fitted GAMLSS model or any other model with suitable standardised residuals.

The function `add_resid_qqplot()` takes a QQ-plot created by `resid_qqplot()` and adds another QQ-plot from a different fitted model.

The function `model_resid_qqplots()` takes different fitted models and creates QQ-plots for all fitted models.

Usage

```
resid_qqplot(obj, resid, value = 3, points.col = "steelblue4",
             line.col = "darkgray", check_overlap = TRUE, title)

add_resid_qqplot(gg, obj, value = 3, points.col = "sienna",
                 line.col = "darkgray", check_overlap = TRUE, title)

model_qqplot(obj, ..., line.col = "steelblue4", title)
```

Arguments

<code>obj</code>	A GAMLSS fitted model or
<code>resid</code>	any other residual suitable standardised.
<code>gg</code>	a ggplot
<code>value</code>	A cut off value to identify large or small residuals

<code>points.col</code>	the colour of the points in the plot
<code>line.col</code>	the colour of the line in the plot
<code>check_overlap</code>	if observations are identify this reduvce the cluterring
<code>title</code>	a title if needed it
<code>...</code>	extra GAMLSS models

Details

This is a stanard QQ-plot but with the advadance of able to identify large or samll residuals

Value

A QQ-plotbis created

Author(s)

Mikis Stasinopoulos, Bob Rigby and Fernanda De Bastiani

References

Rigby, R. A., Stasinopoulos, D. M., Heller, G. Z., and De Bastiani, F. (2019) *Distributions for modeling location, scale, and shape: Using GAMLSS in R*, Chapman and Hall/CRC. An older version can be found in <https://www.gamlss.com/>.

Stasinopoulos D. M. Rigby R.A. (2007) Generalized additive models for location scale and shape (GAMLSS) in R. *Journal of Statistical Software*, Vol. **23**, Issue 7, Dec 2007, <https://www.jstatsoft.org/v23/i07/>.

Stasinopoulos D. M., Rigby R.A., Heller G., Voudouris V., and De Bastiani F., (2017) *Flexible Regression and Smoothing: Using GAMLSS in R*, Chapman and Hall/CRC.

Stasinopoulos, M.D., Kneib, T., Klein, N., Mayr, A. and Heller, G.Z., (2024). *Generalized Additive Models for Location, Scale and Shape: A Distributional Regression Approach, with Applications* (Vol. **56**). Cambridge University Press.

(see also <https://www.gamlss.com/>).

See Also

[plot.gamlss](#)

Examples

```
data(abdom)
a<-gamlss(y~pb(x),family=L0,data=abdom)
b<-gamlss(y~pb(x),family=N0,data=abdom)
gg <- resid_qqplot(a)
add_resid_qqplot(gg, b)
model_qqplot(a,b)
```

resid_symmetry	Symmetry plots
----------------	----------------

Description

The functions `resid_symmetry()` and `y_symmetry()` plot symmetry plots for residuals and single variable, respectively.

Usage

```
resid_symmetry(model, title)
```

```
y_symmetry(y, title)
```

Arguments

<code>model</code>	A model which allows the function <code>resid()</code>
<code>y</code>	a single variable
<code>title</code>	A title for the plot if needed

Details

The function orders the data (or residuals) and takes the median minus the lower half and plot it against the upper half minus the median.

Value

The symmetry plot is produced.

Author(s)

Mikis Stasinopoulos, Bob Rigby and Fernanda De Bastiani

References

Rigby, R. A., Stasinopoulos, D. M., Heller, G. Z., and De Bastiani, F. (2019) *Distributions for modeling location, scale, and shape: Using GAMLSS in R*, Chapman and Hall/CRC. An older version can be found in <https://www.gamlss.com/>.

Stasinopoulos D. M. Rigby R.A. (2007) Generalized additive models for location scale and shape (GAMLSS) in R. *Journal of Statistical Software*, Vol. **23**, Issue 7, Dec 2007, <https://www.jstatsoft.org/v23/i07/>.

Stasinopoulos D. M., Rigby R.A., Heller G., Voudouris V., and De Bastiani F., (2017) *Flexible Regression and Smoothing: Using GAMLSS in R*, Chapman and Hall/CRC.

Stasinopoulos, M.D., Kneib, T., Klein, N., Mayr, A. and Heller, G.Z., (2024). *Generalized Additive Models for Location, Scale and Shape: A Distributional Regression Approach, with Applications* (Vol. **56**). Cambridge University Press.

(see also <https://www.gamlss.com/>).

See Also[resid_index](#)**Examples**

```

y <- rBCT(1000, mu=3, sigma=.1, nu=-1, tau=5)
y_hist(y)
gg <- y_symmetry(y)

```

resid_wp

*Worm plot using ggplot2***Description**

The function produces worm plot of the residuals of a fitted model. A worm plot is a de-trended normal QQ-plot so departure from normality is highlighted.

The function `plot_wp()` it is similar to the **gamlss** package function `wp()` when the argument `xvar` is not used.

Usage

```

resid_wp(obj, resid, value = 3, points_col = "steelblue4",
         poly_col = "darkred",
         check_overlap = TRUE, title, ylim)

model_wp(obj, ..., title)

resid_wp_wrap(obj, resid, value = 3, xvar = NULL, n_inter = 4,
              points_col = "steelblue4", poly_col = "darkred",
              alpha_bound = 0.1, check_overlap = TRUE, title, ylim)

model_wp_wrap(obj, ..., xvar = NULL, value = 3, n_inter = 4,
              points_col = "steelblue4", alpha_bound = 0.1,
              check_overlap = TRUE, ylim, title)

```

Arguments

<code>obj</code>	a GAMLSS fitted object or any other fitted model where the <code>resid()</code> method works (preferably the residuals should be standardised or better normalised quantile residuals. Note for <code>model_wp</code> only <code>gamlss</code> object are accepted.)
<code>resid</code>	if object is missing this argument can be used to specify the residual vector (again it should a normalised quantile residual vector)
<code>value</code>	A cut off point to indicate large residuals, default is <code>value=3</code>
<code>xvar</code>	the x term for which the worm plot will be plotted against
<code>n_inter</code>	the number of intervals for continuous x-term

points_col	the color of the points in the plot
poly_col	the colour of the fitted polynomial in the plot
check_overlap	to check for overlap when plotting the observation numbers
alpha_bound	the transparency parameter for the confidence bound
title	required title
ylim	if the y limit should be different from the default $\max(y)+1$
...	extra GAMLSS models

Value

A worm plot is produced

Author(s)

Mikis Stasinopoulos, Bob Rigby and Fernanda De Bastiani

References

van Buuren and Fredriks M. (2001) Worm plot: simple diagnostic device for modelling growth reference curves. *Statistics in Medicine*, **20**, 1259–1277

Rigby, R. A., Stasinopoulos, D. M., Heller, G. Z., and De Bastiani, F. (2019) *Distributions for modeling location, scale, and shape: Using GAMLSS in R*, Chapman and Hall/CRC. An older version can be found in <https://www.gamlss.com/>.

Stasinopoulos D. M. Rigby R.A. (2007) Generalized additive models for location scale and shape (GAMLSS) in R. *Journal of Statistical Software*, Vol. **23**, Issue 7, Dec 2007, <https://www.jstatsoft.org/v23/i07/>.

Stasinopoulos D. M., Rigby R.A., Heller G., Voudouris V., and De Bastiani F., (2017) *Flexible Regression and Smoothing: Using GAMLSS in R*, Chapman and Hall/CRC.

Stasinopoulos, M.D., Kneib, T., Klein, N., Mayr, A. and Heller, G.Z., (2024). *Generalized Additive Models for Location, Scale and Shape: A Distributional Regression Approach, with Applications* (Vol. **56**). Cambridge University Press.

(see also <https://www.gamlss.com/>).

See Also

[wp](#)

Examples

```
data(abdom)
# with data
a<-gamlss(y~pb(x),sigma.fo=~pb(x,1),family=L0,data=abdom)
resid_wp(a)
resid_wp(resid=resid(a))
```

 resp_mu

Plotting the response against quantities of the fitted model

Description

All plots are of the response variable against fitted values of interest.

The function `resp_mu()` is the ‘original’ one plotting the response against the parameter `mu`. The function reports the Pearson’s correlation coefficient and plot a lines (45% degrees) through the graph.

The function `resp_param()` plots the response against any fitted parameter `mu`, `sigma`, `nu` or `tau`. The function also plots a smooth curve going through the data and gives the Pearson’s correlation coefficient.

The function `resp_quantile()` plots the response against any fitted quantile, with default the median (0.50). The function also plots a smooth curve going through the data and gives the Pearson’s correlation coefficient.

The function `quantile_gamlss()` is used by the function `resp_quantile()` to calculate the quantiles of the fitted distribution .

Usage

```
resp_mu(obj, title, line.col = "darkred",
        point.col = "steelblue4",
        point.shape = 20)

resp_param(obj, param = c("mu", "sigma", "nu", "tau"),
          title, line.col = "darkred", point.col = "steelblue4",
          point.shape = 20)

resp_quantile(obj, quantile = 0.5, title, newdata,
             line.col = "darkred", point.col = "steelblue4",
             point.shape = 20)

quantile_gamlss(obj, quantile = 0.5, newdata)
```

Arguments

<code>obj</code>	a GAMLSS fitted object
<code>param</code>	which parameters? <code>mu</code> , <code>sigma</code> , <code>nu</code> or <code>tau</code> , [only for <code>resp_param()</code>].
<code>quantile</code>	which quantile? default the median (0.50), [only for <code>resp_quantile()</code> and <code>quantile_gamlss()</code>].
<code>title</code>	a title if needed it, by default for the function <code>fitted_resp</code> it print the correlation coefficients between the two variable.
<code>line.col</code>	the colour of the line
<code>point.col</code>	the colour of the points

point.shape	the shape of the points
newdata	whether the evaluation should be in newdata or the old data points [only for functions resp_quantile() and quantile_gamlss()]

Details

This is standard plot in regression where the fitted values are plotted against the response. In GAMLSS model is done by plotting the response against the fitted values of the mu model which is most case is a location parameter.

Value

A plot is returned

Note

Do not use this plot if mu is not a location parameter.

Author(s)

Mikis Stasinopulos, Rober Rigby and Fernanda de Bastiani

References

Rigby, R. A. and Stasinopoulos D. M. (2005). Generalized additive models for location, scale and shape,(with discussion), *Appl. Statist.*, **54**, part 3, pp 507-554.

Rigby, R. A., Stasinopoulos, D. M., Heller, G. Z., and De Bastiani, F. (2019) *Distributions for modeling location, scale, and shape: Using GAMLSS in R*, Chapman and Hall/CRC. An older version can be found in <https://www.gamlss.com/>.

Stasinopoulos D. M. Rigby R.A. (2007) Generalized additive models for location scale and shape (GAMLSS) in R. *Journal of Statistical Software*, Vol. **23**, Issue 7, Dec 2007, <https://www.jstatsoft.org/v23/i07/>.

Stasinopoulos D. M., Rigby R.A., Heller G., Voudouris V., and De Bastiani F., (2017) *Flexible Regression and Smoothing: Using GAMLSS in R*, Chapman and Hall/CRC.

Stasinopoulos, M.D., Kneib, T., Klein, N., Mayr, A. and Heller, G.Z., (2024). *Generalized Additive Models for Location, Scale and Shape: A Distributional Regression Approach, with Applications* (Vol. **56**). Cambridge University Press.

(see also <https://www.gamlss.com/>).

See Also

[resid_plots](#)

Examples

```
m1 <- gamlss(R~pb(F1)+pb(A)+H+loc, data=rent, family=GA )
resp_mu(m1)
resp_param(m1)
resp_quantile(m1)
```

xy_median*The Median in Two Dimensions*

Description

The function `xy_median(x, y)` uses the functions `median_cw(x, y)`, `median_pc(x, y)` and `median_sp(x, y)` to calculate the component wise, principle componet and spatial median, respectively, in 2-Dimensions.

Usage

```
xy_median(x, y, method = c("cw", "pc", "sp"))
```

```
median_cw(x, y)
```

```
median_pc(x, y)
```

```
median_sp(x, y)
```

Arguments

x	the first variable
y	the second variable
method	The choises are <code>c("cw", "pc", "sp")</code> for component wise, principle componet and spatial median, respectively.

Details

The function `xy_median(x, y)` uses the other three functions to calculate the mediam in 2-Dimentionts.

Value

The coordinats of the median

Note

The median in two dimensions need two pieces of information e.g. two coordinates or one cooridate and one direction.

Author(s)

Mikis Stasinopoulos

References

Rigby, R. A., Stasinopoulos, D. M., Heller, G. Z., and De Bastiani, F. (2019) *Distributions for modeling location, scale, and shape: Using GAMLSS in R*, Chapman and Hall/CRC. An older version can be found in <https://www.gamlss.com/>.

Stasinopoulos D. M. Rigby R.A. (2007) Generalized additive models for location scale and shape (GAMLSS) in R. *Journal of Statistical Software*, Vol. **23**, Issue 7, Dec 2007, <https://www.jstatsoft.org/v23/i07/>.

Stasinopoulos D. M., Rigby R.A., Heller G., Voudouris V., and De Bastiani F., (2017) *Flexible Regression and Smoothing: Using GAMLSS in R*, Chapman and Hall/CRC.

Stasinopoulos, M.D., Kneib, T., Klein, N., Mayr, A. and Heller, G.Z., (2024). *Generalized Additive Models for Location, Scale and Shape: A Distributional Regression Approach, with Applications* (Vol. **56**). Cambridge University Press.

(see also <https://www.gamlss.com/>).

See Also

[median](#)

Examples

```
library(gamlss)
median_cw(rent99$rent, rent99$rentsqm)
median_pc(rent99$rent, rent99$rentsqm)
median_sp(rent99$rent, rent99$rentsqm)
xy_median(rent99$rent, rent99$rentsqm)
```

xy_scatter

Data Plot for Bivariate Vectors

Description

This set of plots are for helping visuallise relationship between two continuous variables in the case where both variables could be considered as possible responses. The idea is to investigate graphically their relationship for a possible use of copula.

There are two types of functions. The ones which are operated by `_` e.g. `xy_scatter()` which are using the `ggplot2` package graphics and the ones using ordinary R graphics which are using a point, `.`, as a separator e.g. `xy.ECDF.contour()`. Here we enumerate the `ggplot2` function while the ordinary graphic functions are indicated within.

i) The function `xy_scatter()` plot the scatter plot of the two continuous variables but also add a) the points of the medians, b) the heigh density region (50%) and c) a 2- dimnesional kernal density estimator (KDE) contour plot.

ii) The function `xy_pit_scatter()` is similar to the the function `xy_scatter()` but instead of the original variables it plots their PIT (probability integral trnsormation) values.

iii) The function `xy_hist()` (which convey similar information to `thexy_scatter()`), it plots a 2D-histogram of the original bivariate data. It uses as grid $bins \times bins$. The default value of the option `bins` is $bins = 50$.

iv) The function `xy_hist_kde2d()`, similar information to `xy_hist()`. It is using the MASS package function `kde2d()` to produce the two dimensional histogram.

v) The function `xy_pit_hist()` (similar information to `thexy_pit_scatter()`), it plots a 2D-histogram of the transformed data ia a grid $bins \times bins$, by default $bins = 50$.

vi) The function `xy_density()` produce a kernel density estimator, KDE, od the two continuous variables.

vii) The function `xy_density_plain()` produces identical results to `xy_density()` but it shows just the contour plots of the 2-D kernel density estimator.

viii) The functions `xy_ECDF_contour()` produce a contour plot of the ECDF of x and y . The 2D-ECDF is calculate using the function `ECDF_2D()`.

ix) The function `xy_ECDF_hist()` produce a a 2D-histogram, of the ECDF of x and y . The 2D-ECDF is calculated by the function `ECDF_2D()`.

Note that similar information like the function `xy_ECDF_contour()` can be produve uding the R graphics functions `xy_ECDF.contour()` and `xy_ECDF.image()` and `xy_ECDF.persp()`

Usage

```
xy_scatter(x, y = NULL, method = c("cw", "pc", "sp"),
           density = TRUE, HDR = TRUE,
           density.colour = "lightblue",
           median.colour = "red", median.size = 3,
           point.size = 1, point.colour = "grey",
           HDR.colour = "darkred")
```

```
xy_pit_scatter(x, y = NULL, method = c("cw", "pc", "sp"),
              density = TRUE, HDR = TRUE,
              density.colour = "lightblue",
              median.colour = "red", median.size = 3,
              point.size = 1, point.colour = "grey",
              HDR.colour = "darkred")
```

```
xy_hist(x, y, colour = hcl.colors(20),
        points = TRUE,
        point.colour = "gray",
        point.size = 0.1,
        bins = 50)
```

```
xy_hist_kde2d(x, y, colour = hcl.colors(20),
              points = TRUE, point.colour = "gray",
              point.size = 0.1, bins = 50)
```

```
xy_pit_hist(x, y, colour = hcl.colors(20),
            points = TRUE,
            point.colour = "gray",
```

```
        point.size = 0.1,
        bins = 50)

xy_density(x, y, colour = hcl.colors(20),
           points = FALSE, point.colour = "black",
           point.size = 0.1)

xy_density_plain(x, y, colour = hcl.colors(20),
                points = FALSE, point.colour = "black",
                point.size = 0.1)

xy_pit_density(x, y = NULL,
               points = FALSE,
               point.size = 1, point.colour = "grey")

xy_ECDF_contour(x, y)

xy_ECDF_hist(x, y)

ECDF_2D(value.x, value.y, data.x, data.y)

xy_ECDF.persp(x, y, theta = 30, phi = 30, xlab = "x",
              ylab = "y", zlab = "f(x,y)",
              colour = "lightblue", main = "ECDF(x,y)")

xy_ECDF.contour(x, y, colour = hcl.colors(20),
               main = "Empirical CDF Heatmap",
               points = TRUE, point.colour = "gray",
               point.size = 0.1)

xy_ECDF.image(x, y, colour = hcl.colors(20),
              main = "Empirical CDF Heatmap",
              contour = TRUE, points = TRUE,
              point.colour = "gray", point.size = 0.1)

xy_density.contour(x, y, colour = hcl.colors(20),
                  main = "Empirical Density", points = TRUE,
                  point.colour = "gray", point.size = 0.1, bins = 50)

xy_density.image(x, y, colour = hcl.colors(20),
                 main = "Empirical Density", contour = TRUE,
                 points = TRUE, point.colour = "gray",
                 point.size = 0.1, bins = 50)

xy_density.persp(x, y, bins = 25,
                 main = "Empirical Bivariate Density")
```

Arguments

x	The first continuous variable
y	The second continuous variable
method	The median method: i) cw; case wise, pc; ii) principal componet, iii) sp spacial
density	whether to plot the density
HDR	whether to plot the hight density region HDR
density.colour	The colour of the kernal density estimator
median.colour	the colour for the point for the median
median.size	The size of the point for the mediam
point.size	The size of the points points for the scatter plot
point.colour	the colour of the points in he scatter plot
HDR.colour	the colour used for the HDR
colour	the colour of the respective plot
points	whether to show the points
bins	the number of the bins in the the 2-D histogram defaul 50
value.x	the value of x to evaluate the ECDF
value.y	the value of y to evaluate the ECDF
data.x	the x data vector for the creation of the ECDF
data.y	the y data vector for the creation of the ECDF
theta, phi	angles defibne the view direction
xlab	the label for x
ylab	the label for y
zlab	the label for z
main	the title of the plot
contour	whether contour

Details

This set of plots, is, to help to discover relationship between two continuous response variables.

Value

Thew result is a relevant plot

Author(s)

Mikis Stasinopoulos

References

Rigby, R. A., Stasinopoulos, D. M., Heller, G. Z., and De Bastiani, F. (2019) *Distributions for modeling location, scale, and shape: Using GAMLSS in R*, Chapman and Hall/CRC. An older version can be found in <https://www.gamlss.com/>.

Stasinopoulos D. M. Rigby R.A. (2007) Generalized additive models for location scale and shape (GAMLSS) in R. *Journal of Statistical Software*, Vol. **23**, Issue 7, Dec 2007, <https://www.jstatsoft.org/v23/i07/>.

Stasinopoulos D. M., Rigby R.A., Heller G., Voudouris V., and De Bastiani F., (2017) *Flexible Regression and Smoothing: Using GAMLSS in R*, Chapman and Hall/CRC.

Stasinopoulos, M.D., Kneib, T., Klein, N., Mayr, A. and Heller, G.Z., (2024). *Generalized Additive Models for Location, Scale and Shape: A Distributional Regression Approach, with Applications* (Vol. **56**). Cambridge University Press.

(see also <https://www.gamlss.com/>).

See Also

[xy_median](#)

Examples

```
library(gamlss)
x <- rent99$rent
y <- rent99$rentsqm
xy_scatter(x,y)
xy_pit_scatter(x,y)
xy_hist(x,y)
xy_pit_hist(x,y)
xy_density(x,y)
xy_density_plain(x,y)
xy_pit_density(x,y)
xy_ECDF_hist(x,y)
xy_ECDF_contour(x,y)
# for pit ECDF use
n <- length(x)
u <- rank(x) / (n + 1)
v <- rank(y) / (n + 1)
xy_ECDF_hist(u,v)
xy_ECDF_contour(u,v)
```

Description

The function `y_hist()` creates a histogram and a density plot for a continuous variable.

The functions `y_acf()` and `y_pacf()` plot the autocorrelation and partial autocorrelation functions for `y`.

The function `y_dots()` is design for highly skewed variables. It is a plot emphasising long tails of the distribution for such variables.

Usage

```
y_hist(y, data, xlab = NULL, with.density = TRUE, hist.col = "black",
       hist.fill = "white", dens.fill = "#FF6666",
       binwidth = (max(y)-min(y))/20, from, to, title)
```

```
y_acf(x, data, title)
```

```
y_pacf(x, data, title)
```

```
y_dots(y, data, value=3, point.size = 2, point.col = "gray",
       quantile = c(.10, .50, .90),
       line.col = c("black","red", "black"),
       line.type = c("dotted", "solid", "dotted"),
       line.size = c(1,1,1), x.axis.col = "black",
       x.axis.line.type = "solid", seed = 123, from, to, title,
       plot = TRUE)
```

Arguments

<code>y, x</code>	a continuous variable
<code>data</code>	where to find argument <code>y</code>
<code>xlab</code>	The x-label if it should be different from the variable name
<code>value</code>	value to identify outliers i.e. for upper tail an outliers is if it is greater than $Q_3 + \text{value} * \text{IQ}$
<code>with.density</code>	whether a density is required, default is TRUE
<code>hist.col</code>	the colour of lines of the histogram
<code>hist.fill</code>	the colour of the histogram
<code>dens.fill</code>	the color of the density plot
<code>binwidth</code>	the binwidth for the histogram
<code>from</code>	where to start the histogram (you may have to change binwidth)
<code>to</code>	where to finish the histogram (you may have to change binwidth)
<code>point.size</code>	the size of the points in <code>y_dots</code>
<code>point.col</code>	the colour of the points in <code>y_dots</code>
<code>quantile</code>	the quantile values to plot in <code>y_dots</code> , the default is 0.10, .50 and .90
<code>line.col</code>	the color of the vertical lines indicating the 0.10, .50 and .90 quantiles in <code>y_dots</code>

line.type	the type of the vertical lines indicating the 0.10, .50 and .90 quantiles in y_dots
line.size	the size of the vertical lines indicating the 0.10, .50 and .90 quantiles in y_dots
x.axis.col	the colour of the x-axis
x.axis.line.type	the type of the x-axis
seed	the seed to jitter the y
title	use this for a different title
plot	whether to plot or just print the outliers

Value

A ggplot is returned

Author(s)

Mikis Stasinopoulos, Bob Rigby and Fernanda De Bastiani

References

Rigby, R. A., Stasinopoulos, D. M., Heller, G. Z., and De Bastiani, F. (2019) *Distributions for modeling location, scale, and shape: Using GAMLSS in R*, Chapman and Hall/CRC. An older version can be found in <https://www.gamlss.com/>.

Stasinopoulos D. M. Rigby R.A. (2007) Generalized additive models for location scale and shape (GAMLSS) in R. *Journal of Statistical Software*, Vol. **23**, Issue 7, Dec 2007, <https://www.jstatsoft.org/v23/i07/>.

Stasinopoulos D. M., Rigby R.A., Heller G., Voudouris V., and De Bastiani F., (2017) *Flexible Regression and Smoothing: Using GAMLSS in R*, Chapman and Hall/CRC.

Stasinopoulos, M.D., Kneib, T., Klein, N., Mayr, A. and Heller, G.Z., (2024). *Generalized Additive Models for Location, Scale and Shape: A Distributional Regression Approach, with Applications* (Vol. **56**). Cambridge University Press.

(see also <https://www.gamlss.com/>).

See Also

[plot.ecdf](#)

Examples

```
library(ggplot2)
y <- rBCT(1000, mu=3, sigma=.1, nu=-1, tau=5)
y_hist(y)
gg <- y_hist(y, with.dens=FALSE)
gg + stat_function(fun = dBCT, args=list(mu=3, sigma=.1, nu=-1, tau=5),
                  colour = "black")
gg + stat_function(fun = dBCT, args=list(mu=3, sigma=.1, nu=-1, tau=5),
                  geom = "area", alpha=0.5, fill="pink", color="black", n=301)

y_acf(diff(EuStockMarkets[,1]))
```

```
y_dots(rent$R)
```

Index

- * **GAMLSS**
 - resp_mu, 56
- * **diagnostics**
 - moment_bucket, 32
- * **growth curves**
 - fitted_centiles, 18
- * **model selection**
 - model_GAIC, 28
- * **package**
 - gamlss.ggplots-package, 2
- * **regression**
 - bivar_fun, 4
 - boot_coef, 6
 - centile_bucket, 7
 - family_pdf, 10
 - fit_PB, 13
 - fitted_cdf, 14
 - fitted_centiles, 18
 - fitted_devianceIncr, 21
 - fitted_leverage, 23
 - fitted_terms, 24
 - histSmo_plot, 27
 - model_GAIC, 28
 - model_pca, 31
 - moment_bucket, 32
 - moment_gray_half, 34
 - pcr_coef_path, 36
 - pe_param, 37
 - prof_term, 43
 - resid_density, 45
 - resid_dtop, 46
 - resid_index, 49
 - resid_qqplot, 51
 - resid_symmetry, 53
 - resid_wp, 54
 - resp_mu, 56
 - xy_median, 58
 - xy_scatter, 59
 - y_hist, 63
 - add_resid_qqplot (resid_qqplot), 51
 - bivar.fun.contour (bivar_fun), 4
 - bivar.fun.image (bivar_fun), 4
 - bivar.fun.persp (bivar_fun), 4
 - bivar_fun, 4
 - bivar_fun_plotly (bivar_fun), 4
 - bivar_pdf (bivar_fun), 4
 - boot_coef, 6
 - boot_coef_one (boot_coef), 6
 - centile_bucket, 7
 - centile_bucket_wrap (centile_bucket), 7
 - centile_colour_both (moment_gray_half), 34
 - centile_colour_half (moment_gray_half), 34
 - centile_gray_both (moment_gray_half), 34
 - centiles, 20
 - centileSK, 35
 - ECDF_2D (xy_scatter), 59
 - family_cdf (fitted_cdf), 14
 - family_pdf, 10
 - fit_PB, 13
 - fitted_cdf, 14
 - fitted_cdf_data (fitted_cdf), 14
 - fitted_centiles, 18
 - fitted_centiles_gap (fitted_centiles), 18
 - fitted_centiles_legend (fitted_centiles), 18
 - fitted_devianceIncr, 21
 - fitted_leverage, 23
 - fitted_pdf (family_pdf), 10
 - fitted_pdf_data (family_pdf), 10
 - fitted_survival (fitted_cdf), 14
 - fitted_survival_data (fitted_cdf), 14
 - fitted_terms, 24

- GAIC_scaled, [30](#)
- gamlss, [3](#), [12](#), [17](#), [22](#), [24](#), [32](#), [51](#)
- gamlss.family, [3](#)
- gamlss.ggplots
 - (gamlss.ggplots-package), [2](#)
- gamlss.ggplots-package, [2](#)
- getPEF, [41](#)
- histSmo, [28](#)
- histSmo_plot, [27](#)
- median, [59](#)
- median_cw(xy_median), [58](#)
- median_pc(xy_median), [58](#)
- median_sp(xy_median), [58](#)
- model_cent_bucket(centile_bucket), [7](#)
- model_centiles(fitted_centiles), [18](#)
- model_density(resid_density), [45](#)
- model_devianceIncr_diff
 - (fitted_devianceIncr), [21](#)
- model_GAIC, [28](#)
- model_GAIC_lollipop(model_GAIC), [28](#)
- model_mom_bucket(moment_bucket), [32](#)
- model_pca, [31](#)
- model_qqplot(resid_qqplot), [51](#)
- model_TD(model_GAIC), [28](#)
- model_TD_lollipop(model_GAIC), [28](#)
- model_wp(resid_wp), [54](#)
- model_wp_wrap(resid_wp), [54](#)
- moment_bucket, [32](#)
- moment_bucket_wrap(moment_bucket), [32](#)
- moment_colour_both(moment_gray_half), [34](#)
- moment_colour_half(moment_gray_half), [34](#)
- moment_gray_both(moment_gray_half), [34](#)
- moment_gray_half, [34](#)
- momentSK, [35](#)
- pcr_coef_path, [36](#)
- pcr_path(pcr_coef_path), [36](#)
- pe_1_parameter(pe_param), [37](#)
- pe_2_parameter(pe_param), [37](#)
- pe_eta(pe_param), [37](#)
- pe_eta_grid(pe_param), [37](#)
- pe_param, [37](#), [43](#)
- pe_param_grid(pe_param), [37](#)
- pe_pdf, [41](#)
- pe_pdf_grid(pe_pdf), [41](#)
- pe_terms(fitted_terms), [24](#)
- plot.ecdf, [65](#)
- plot.gamlss, [46](#), [51](#), [52](#)
- plot.PCR, [37](#)
- predict_cdf(fitted_cdf), [14](#)
- predict_pdf(family_pdf), [10](#)
- prof.term, [45](#)
- prof_term, [43](#)
- Q.stats, [10](#), [34](#)
- quantile_gamlss(resp_mu), [56](#)
- resid_density, [12](#), [45](#)
- resid_dtop, [46](#)
- resid_ecdf(resid_dtop), [46](#)
- resid_index, [32](#), [49](#), [54](#)
- resid_median(resid_index), [49](#)
- resid_mu(resid_index), [49](#)
- resid_param(resid_index), [49](#)
- resid_plots, [57](#)
- resid_plots(resid_index), [49](#)
- resid_qqplot, [51](#)
- resid_quantile(resid_index), [49](#)
- resid_symmetry, [53](#)
- resid_wp, [48](#), [54](#)
- resid_wp_wrap(resid_wp), [54](#)
- resid_xvar(resid_index), [49](#)
- resp_mu, [56](#)
- resp_param(resp_mu), [56](#)
- resp_quantile(resp_mu), [56](#)
- term.plot, [7](#), [26](#)
- wp, [10](#), [34](#), [55](#)
- xy.density.contour(xy_scatter), [59](#)
- xy.density.image(xy_scatter), [59](#)
- xy.density.persp(xy_scatter), [59](#)
- xy.ECDF.contour(xy_scatter), [59](#)
- xy.ECDF.image(xy_scatter), [59](#)
- xy.ECDF.persp(xy_scatter), [59](#)
- xy_density(xy_scatter), [59](#)
- xy_density_plain(xy_scatter), [59](#)
- xy_ECDF_contour(xy_scatter), [59](#)
- xy_ECDF_hist(xy_scatter), [59](#)
- xy_hist(xy_scatter), [59](#)
- xy_hist_kde2d(xy_scatter), [59](#)
- xy_median, [58](#), [63](#)
- xy_pit_density(xy_scatter), [59](#)

`xy_pit_hist(xy_scatter)`, [59](#)
`xy_pit_scatter(xy_scatter)`, [59](#)
`xy_scatter`, [59](#)

`y_acf(y_hist)`, [63](#)
`y_dots(y_hist)`, [63](#)
`y_ecdf(resid_dtop)`, [46](#)
`y_epdf(resid_dtop)`, [46](#)
`y_epit(resid_dtop)`, [46](#)
`y_hist`, [63](#)
`y_pacf(y_hist)`, [63](#)
`y_symmetry(resid_symmetry)`, [53](#)